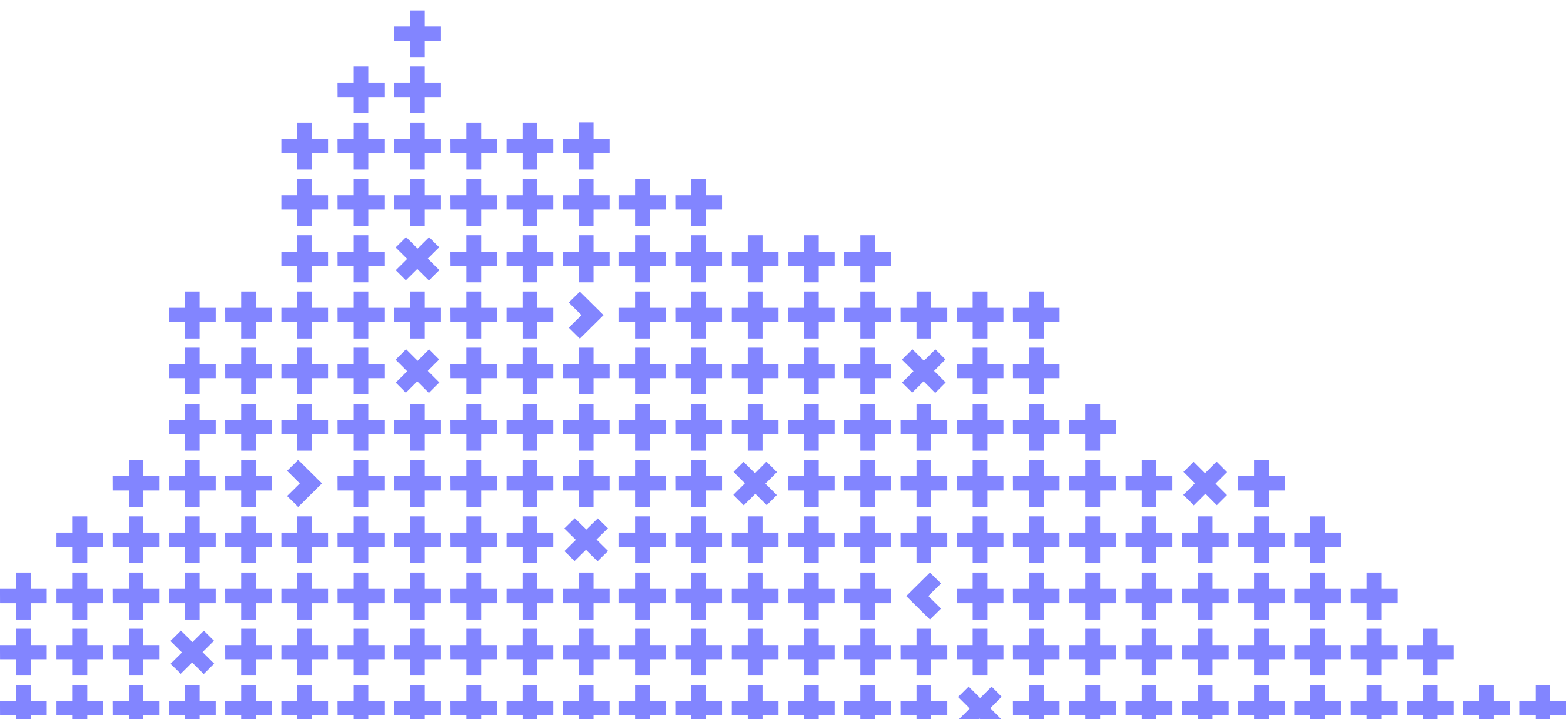


# Reducing traffic wastage in video player

Olga Popova



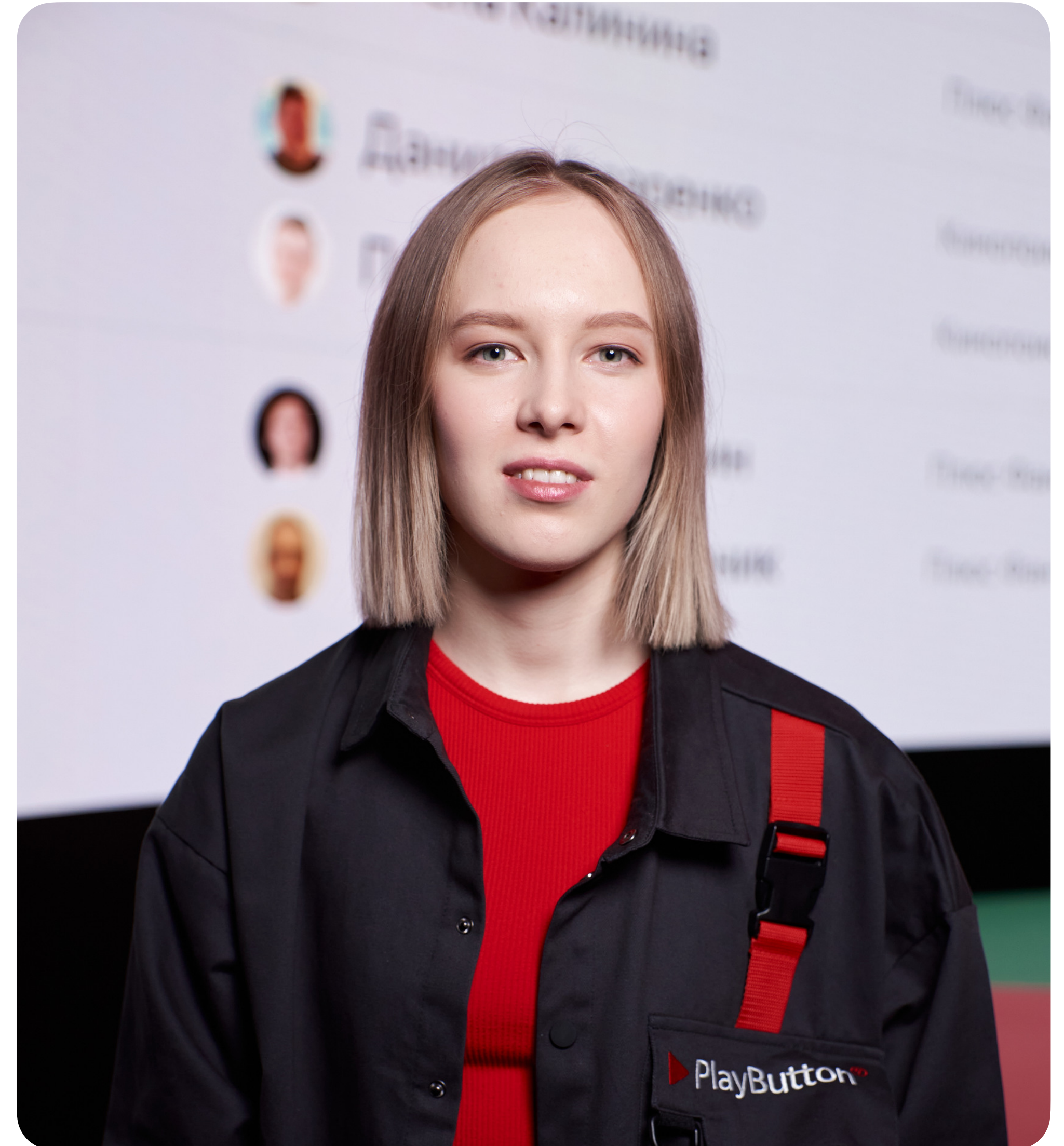
Co-organizer

**Yandex**

# Olga Popova

- web video player team
- ABR, QoE, startup time
- talk about ABR on VideoTech 2021

UI developer in Yandex Cloud





# Agenda

## 1. Theory

- How do we lose traffic?
- On what video player aspects can we influence?
- QoE vs reducing data wastage
- How do we connect traffic with product metrics?

## 2. The evolution of the reducing traffic KPI-metric

## 3. Harsh reality

- Buffer limit to X seconds
- Dynamic buffer
- Skippable fragments
- Viewport capping
- Aesthete capping
- SwitchUp capping

## 4. Results and conclusions

# What problem are we solving?

Decrease costs on the traffic delivery



This talk is about the  
reducing the traffic wastage  
from a client



Theory. How to  
reduce the data  
wastage from client?



How do we lose ~~traffic~~?  
money

# Main data wastage scenarios

- Wrong capping
- Loss of the data we've already downloaded



# Wrong ~~capping~~ cupping





**cap \ 'kap \**

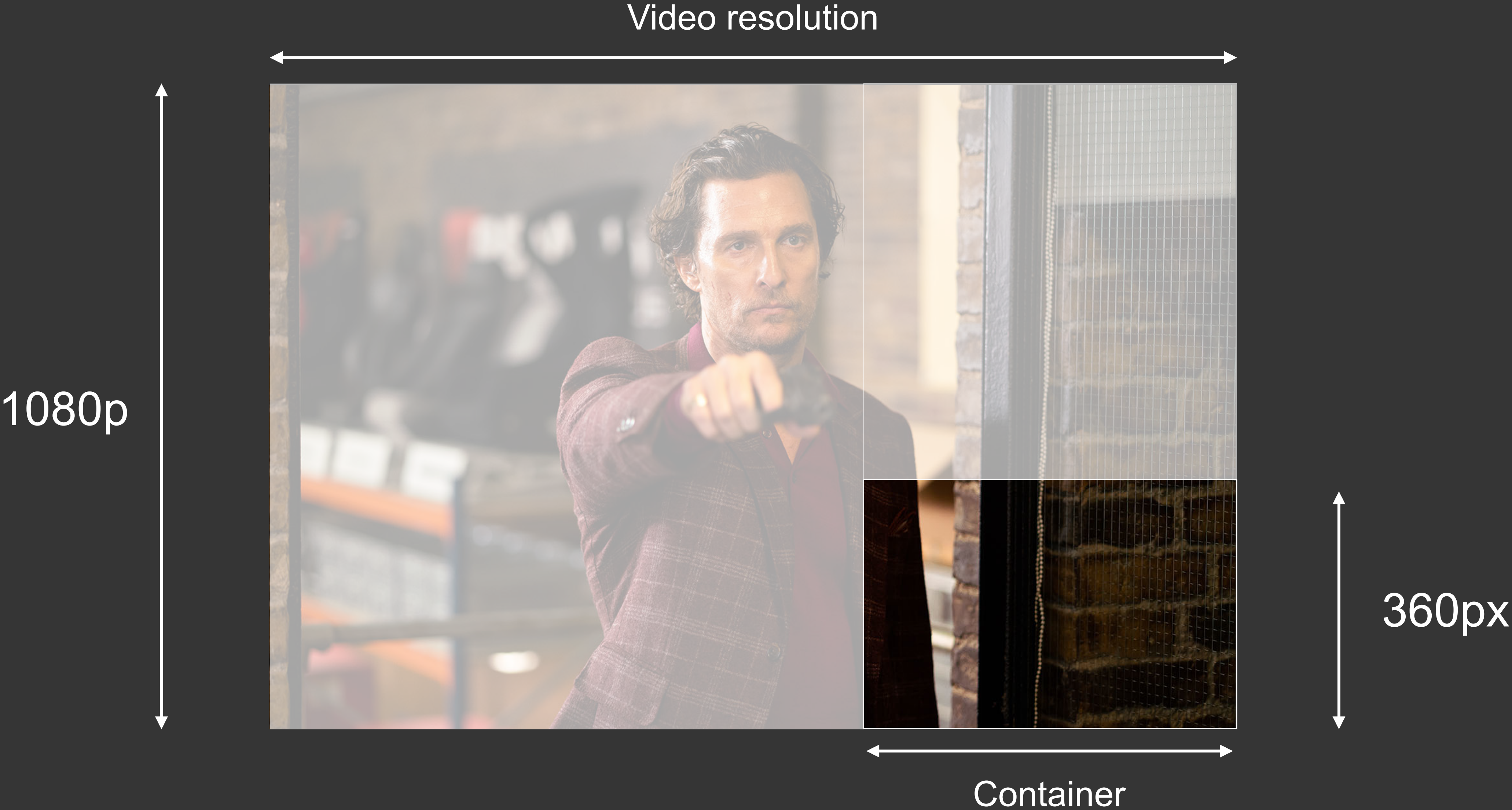
*(verb)*

**: to prevent from growing or  
spreading : set an upper limit on**

*// cap oil prices*



# Capping



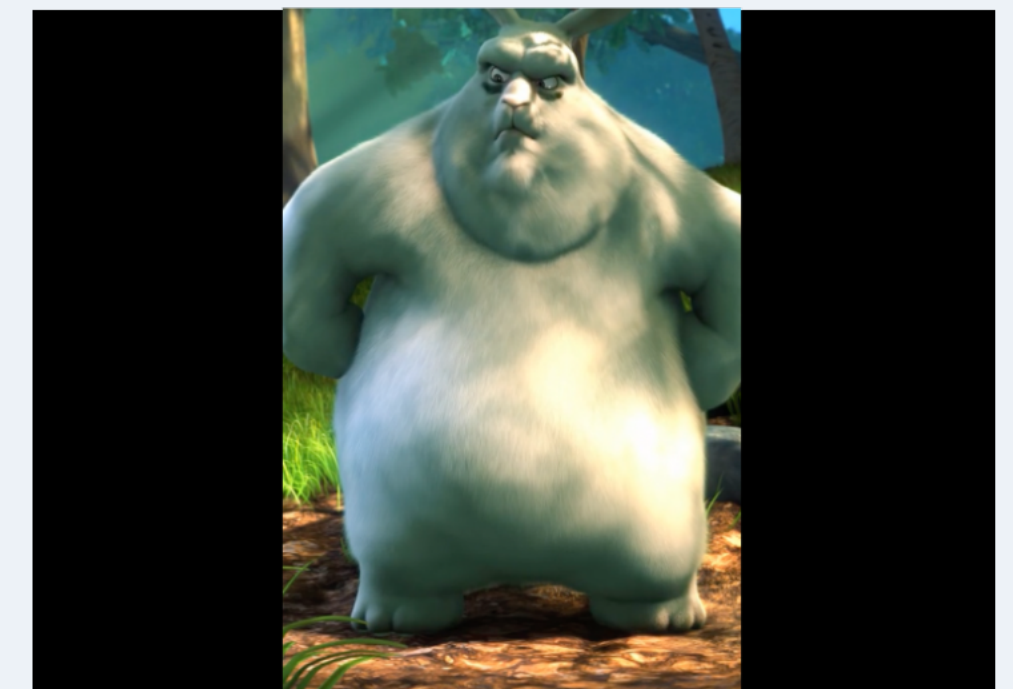
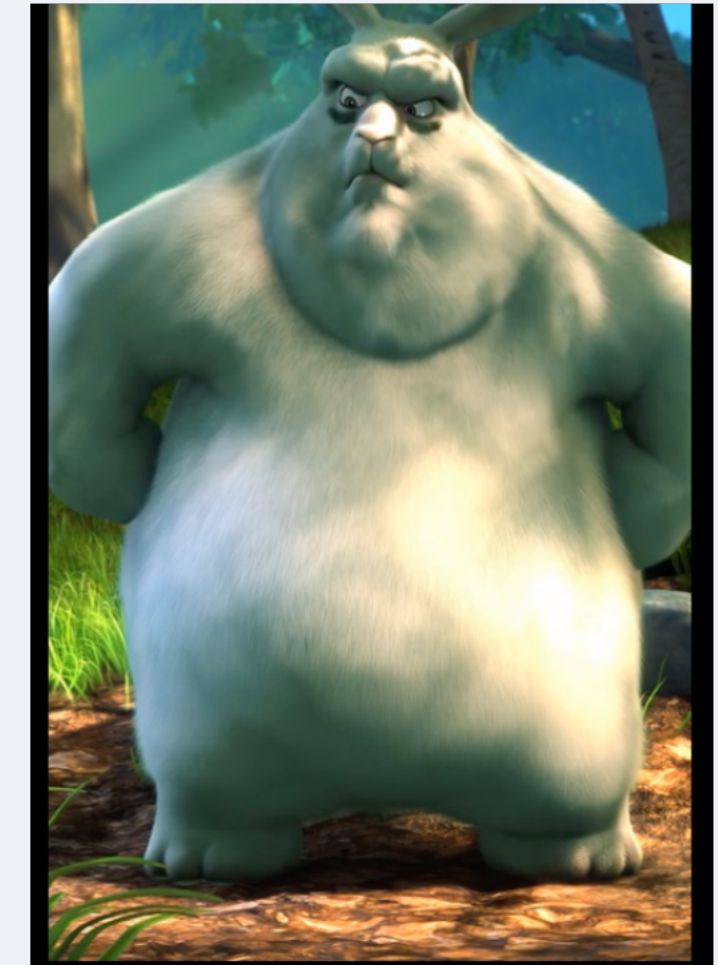


# Capping

Capping - limiting the qualities array according to the container dimensions

video  
orientation

container orientation



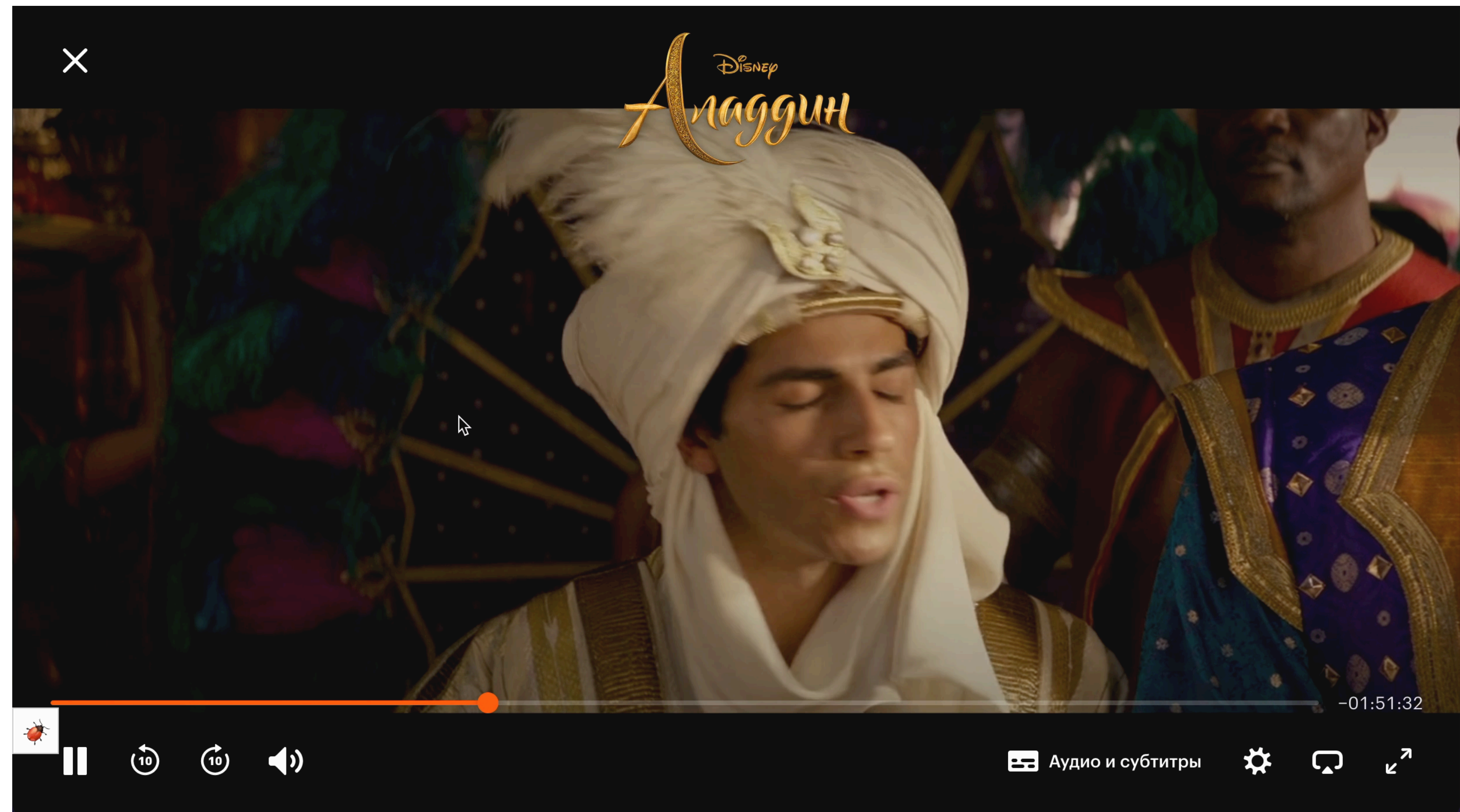
# Loss of the data we've already downloaded

- Early departure
- Video skip (seeking)
- Quality switch by user



# Loss of the data we've already downloaded

- **Early departure**
- Video skip (seeking)
- Quality switch by user

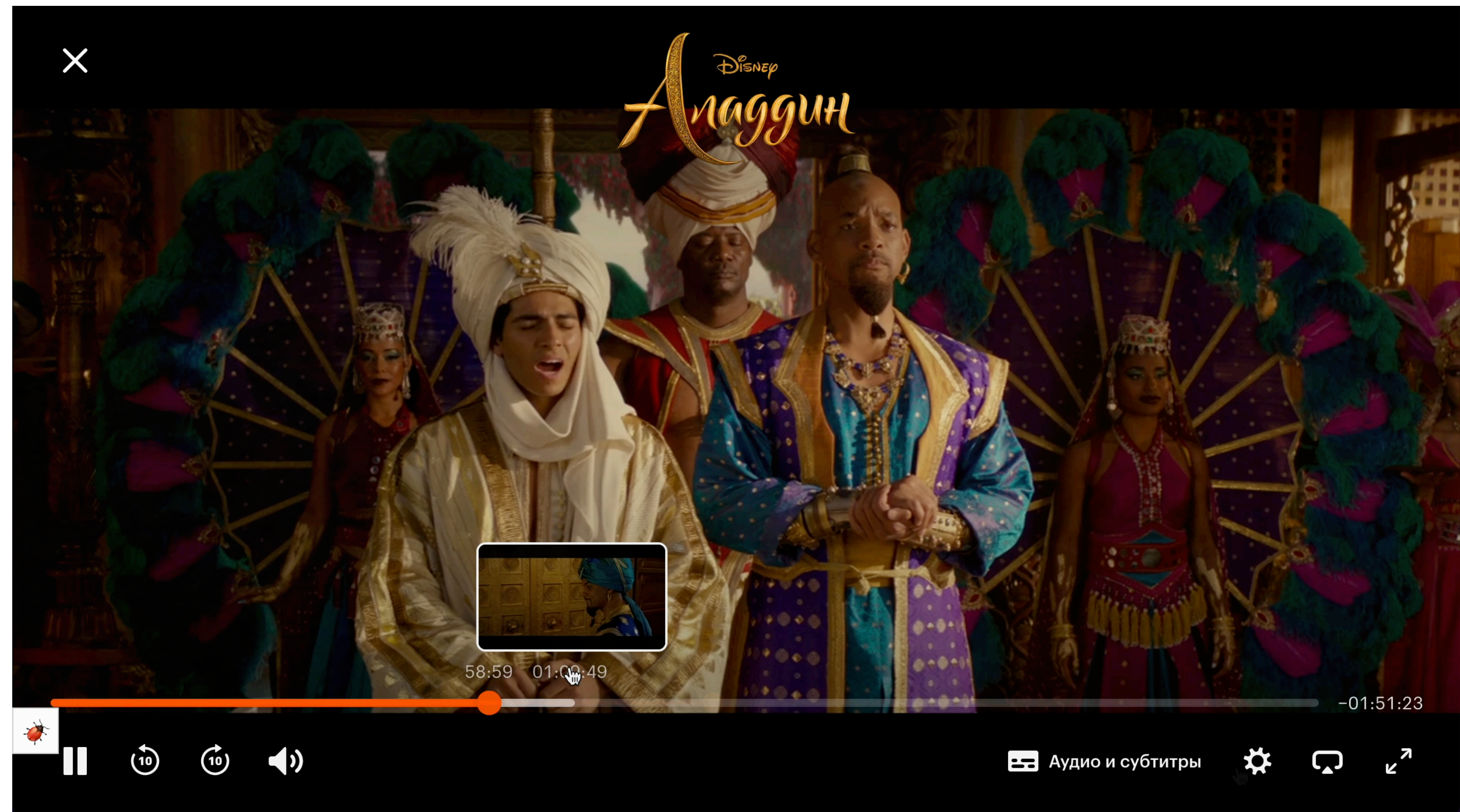




## Theory

# Loss of the data we've already downloaded

- Early departure
- **Video skip (seeking)**
- Quality switch by user

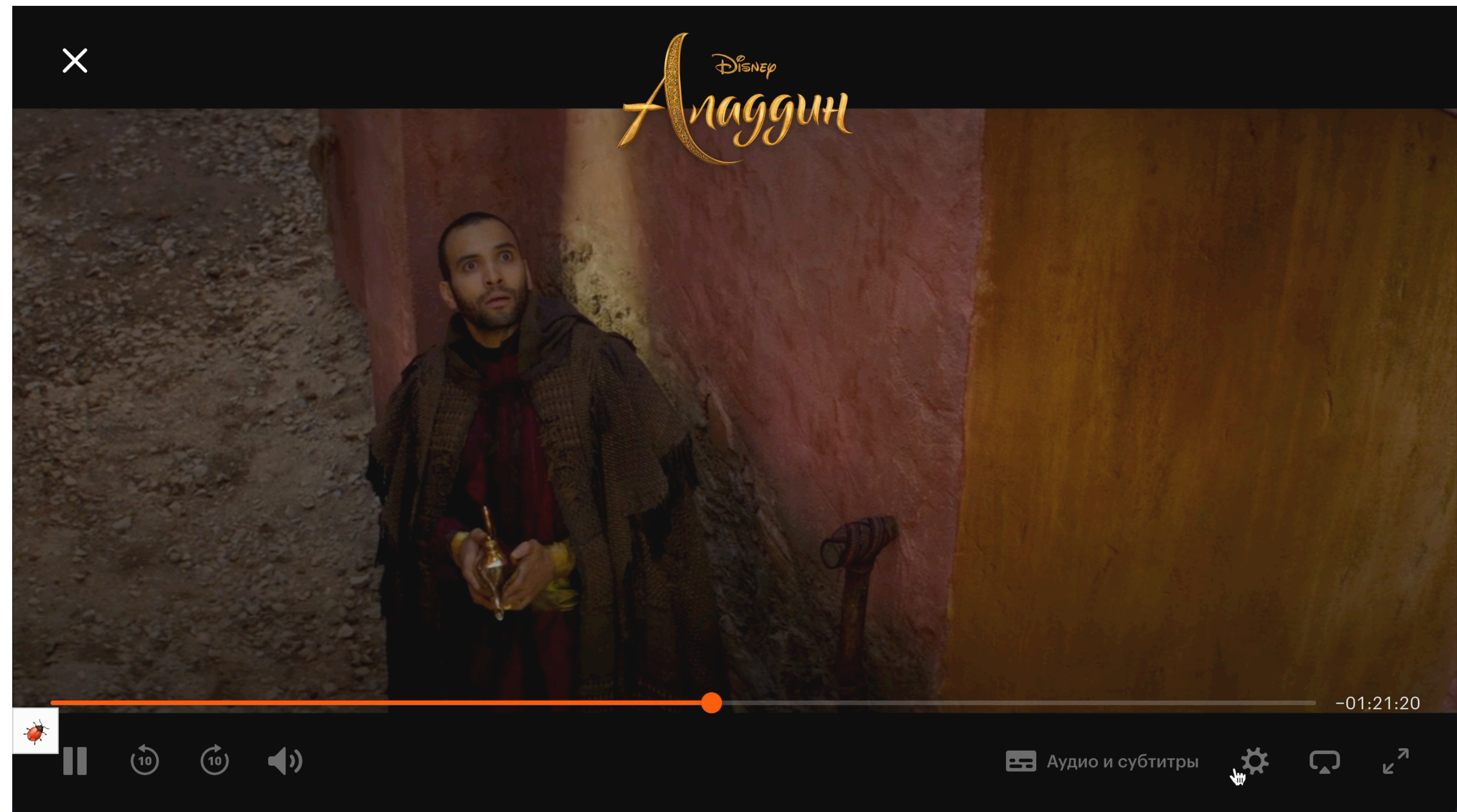




## Theory

# Loss of the data we've already downloaded

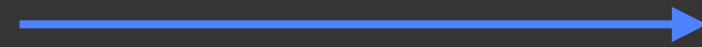
- Early departure
- Video skip (seeking)
- **Quality switch by user**



# « ABR - adaptive bitrate

# What video player aspects can we influence?

- Buffer size
- Video quality



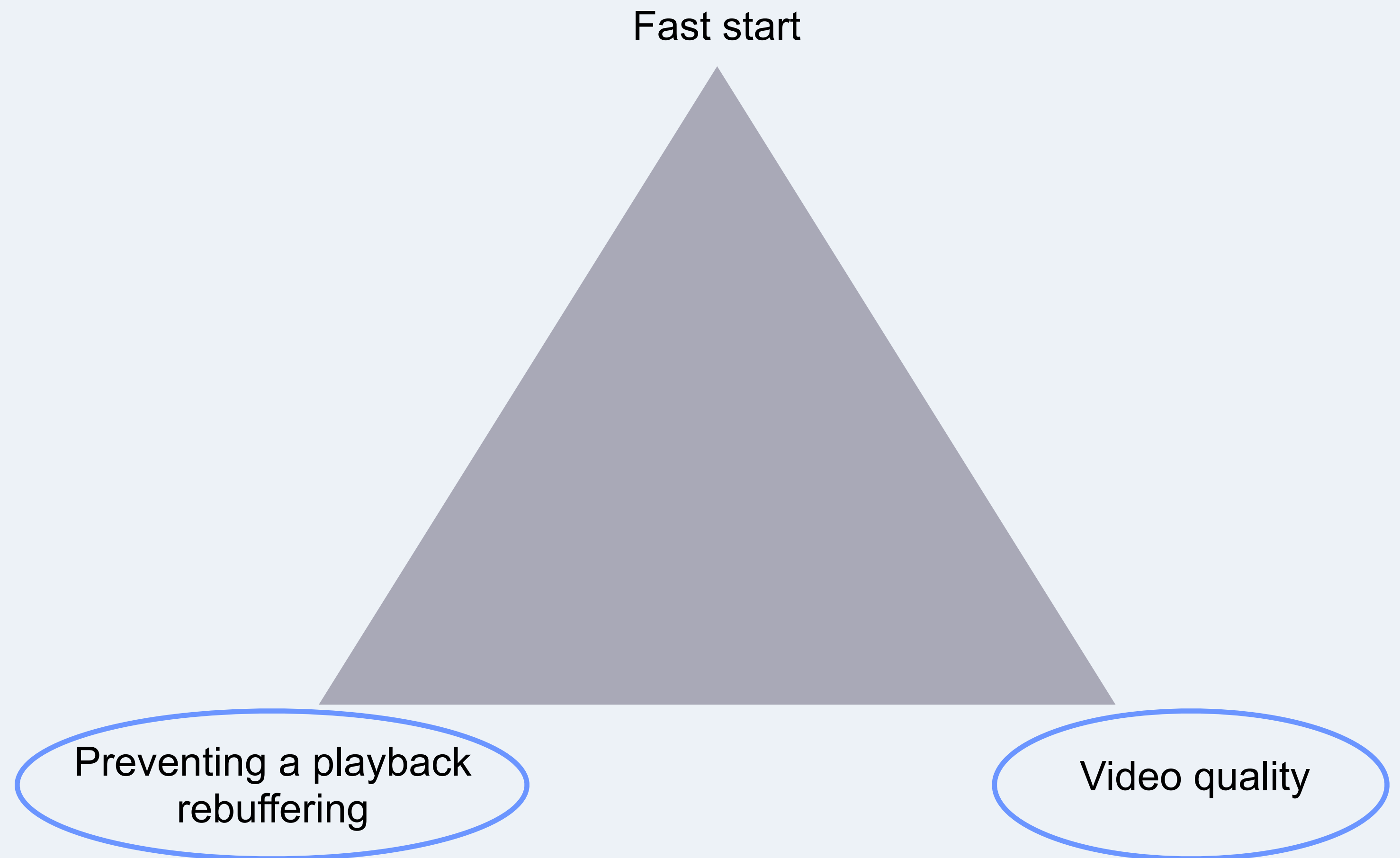
- **Decrease** the buffer size
- **Decrease** the video quality

# QoE vs reducing the traffic wastage

## Quality of experience

is a measure of the delight or annoyance of a customer's experiences with a service

*(Wikipedia)*







Reducing the  
traffic wastage

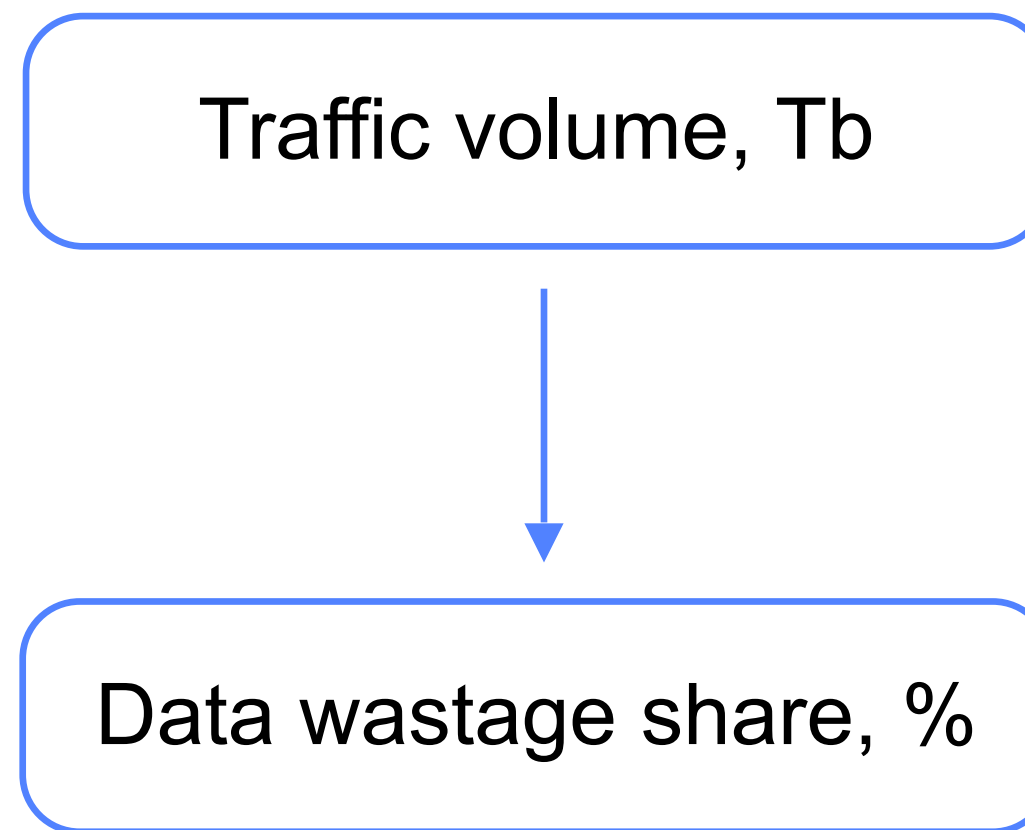
QoE

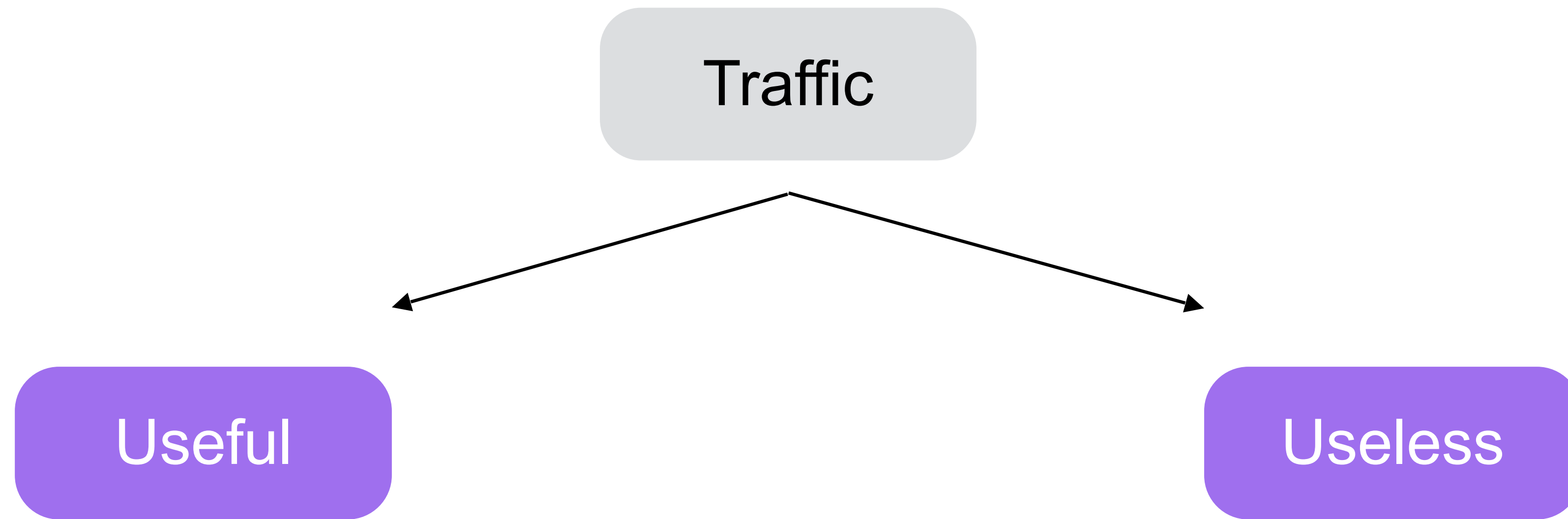


# The evolution of the reducing the traffic KPI-metric



# The evolution of the reducing the traffic KPI-metric



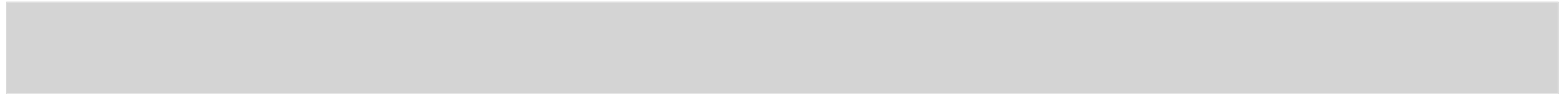


Which was watched by user

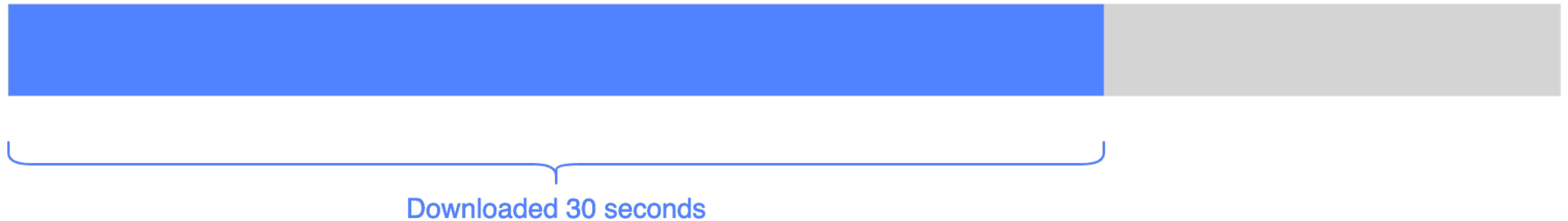
Which **wasn't** watched by user



# Data wastage share

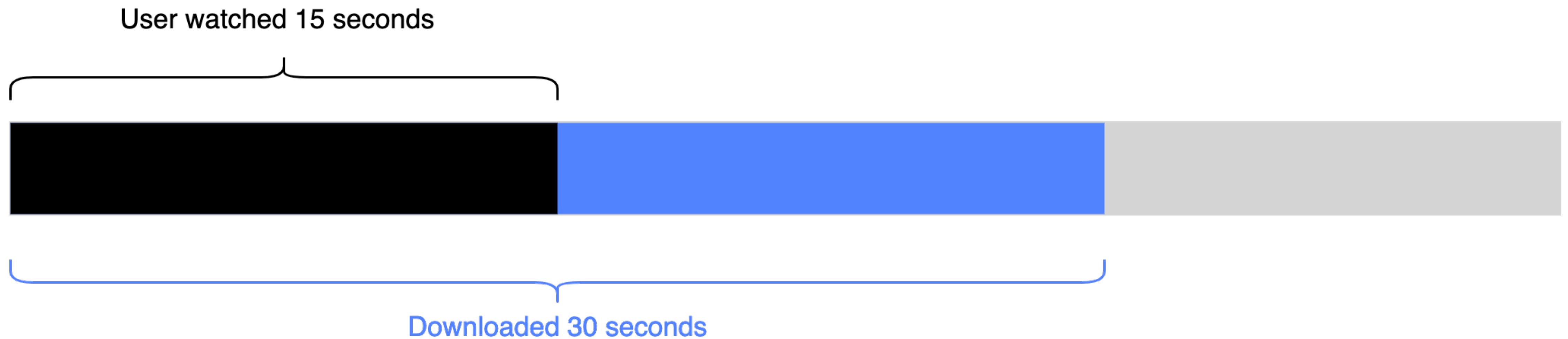


# Data wastage share





# Data wastage share



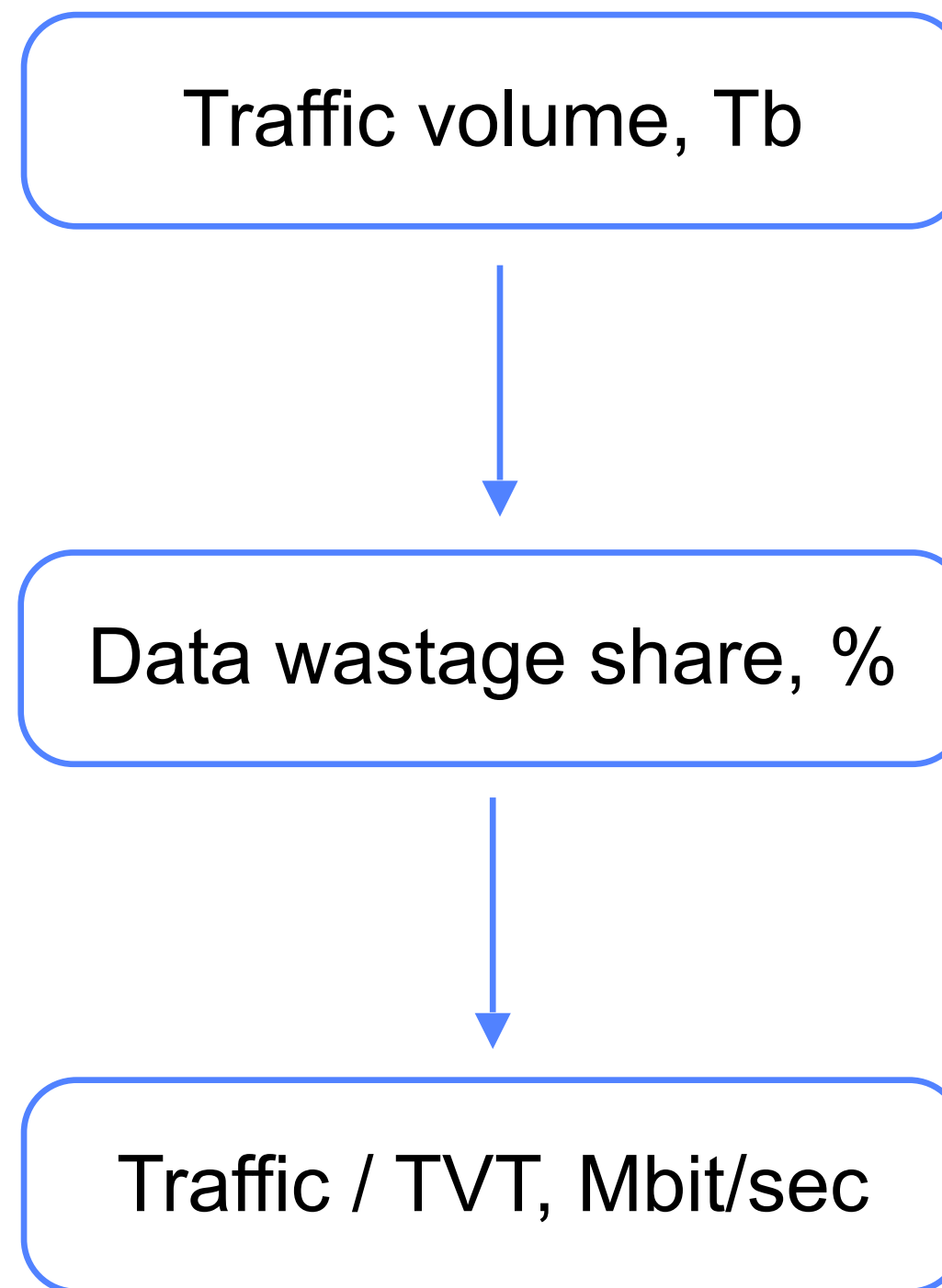
# Data wastage share



Share of a viewed content in seconds  $\neq$  Share useful traffic



# The evolution of the reducing the traffic KPI-metric



TVT (Total View Time)

# Traffic / TVT

👍 Takes into account the growth of the service

👎 Doesn't take into account hype 4k titles

👎 Non-obvious metric



# Harsh reality

# Our hypotheses

## **Decrease the buffer size**

Buffer limit to X seconds

Dynamic buffer

Skippable fragments

## **Decrease the video quality**

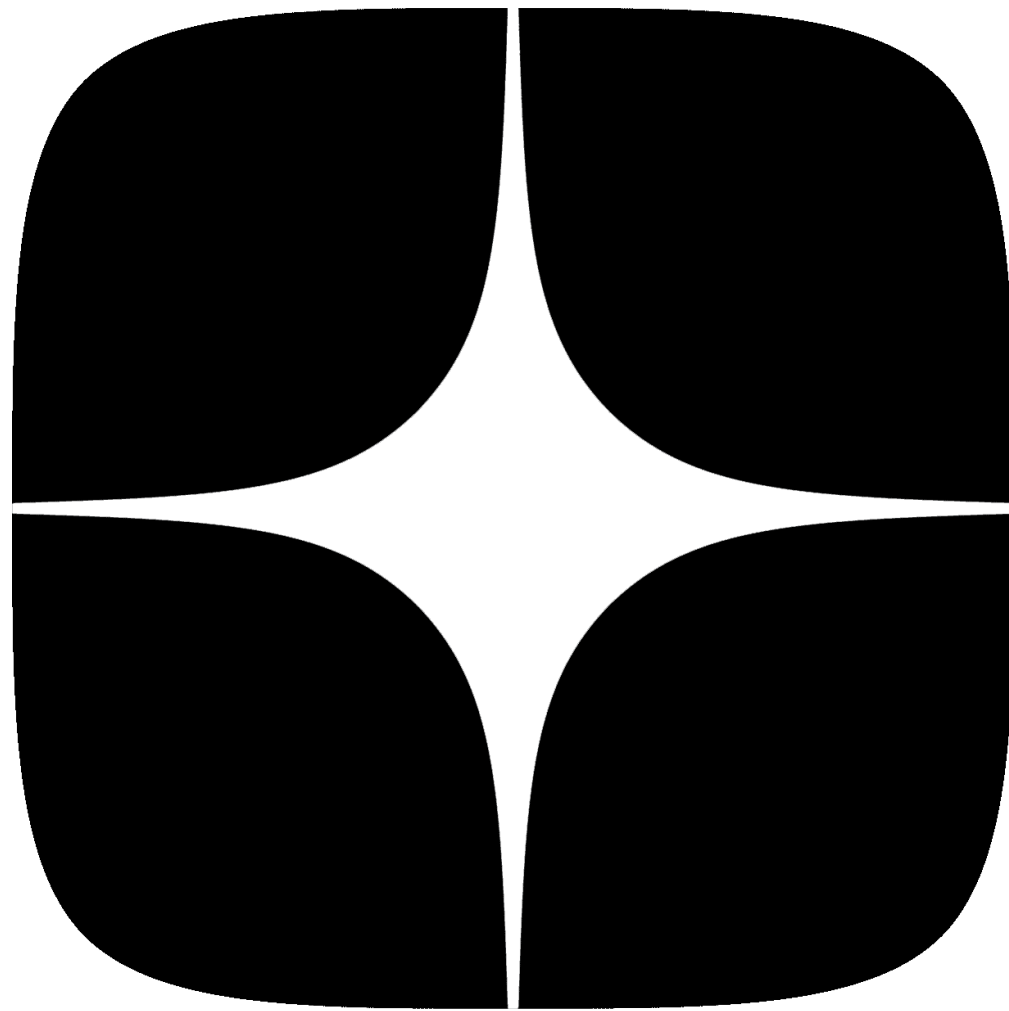
Viewport capping

Aesthete capping

SwitchUp capping



# Apps where we had experiments



Zen



Kinopoisk

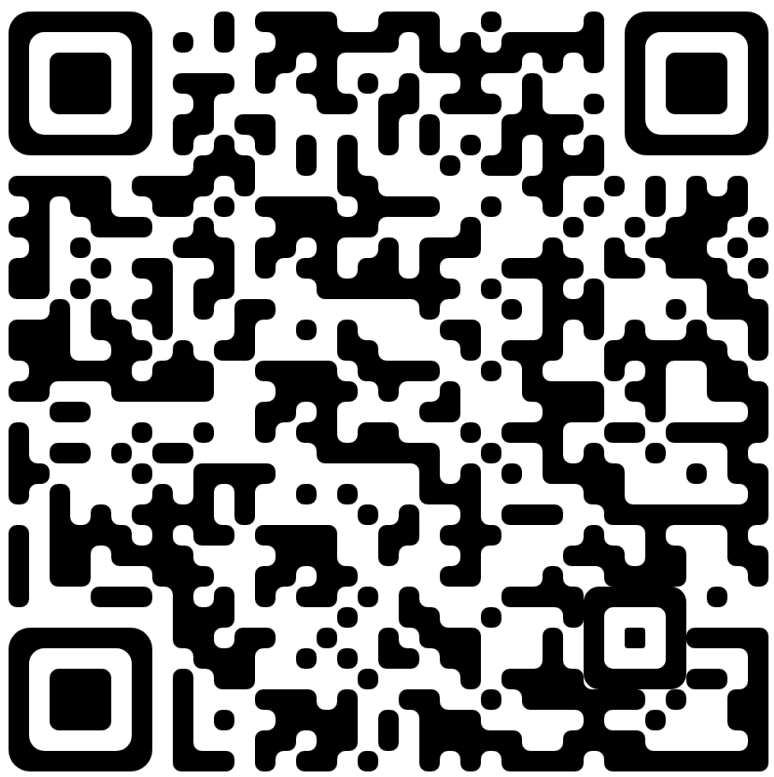
Buffer limit to X  
seconds



# Buffer limit to X seconds: web

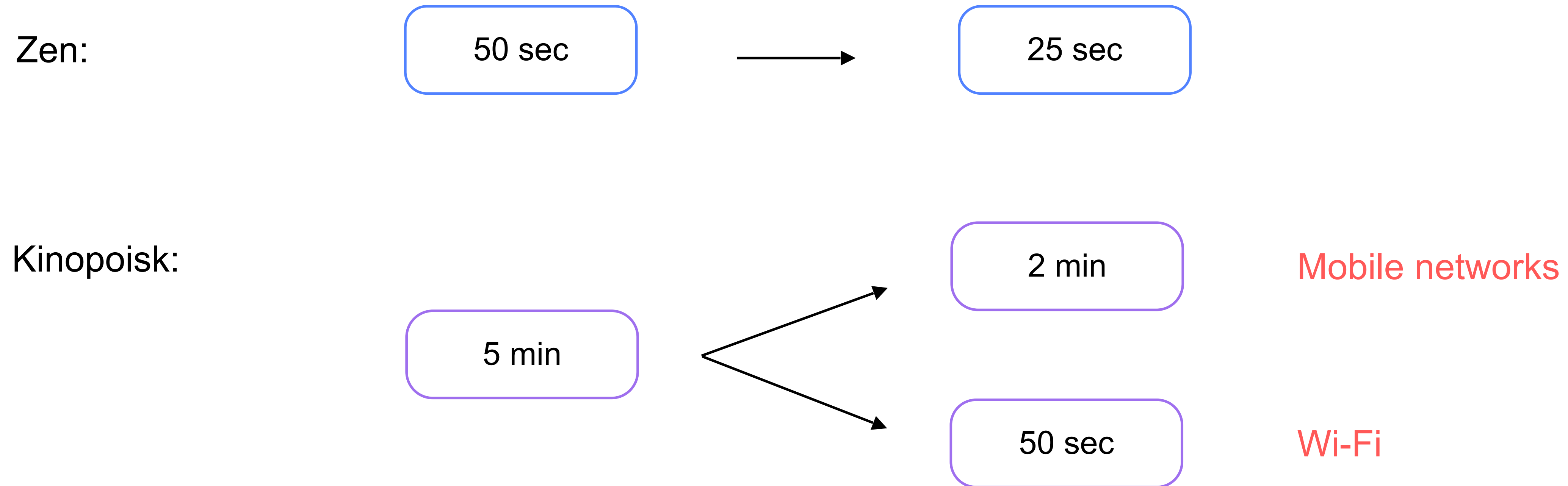
30 second limit

	Chrome	Chromecast*	Firefox	Safari
Video	150MB	30MB	100MB	290MB
Audio	12MB	2MB	15MB	14MB



<https://clck.ru/32qZqy>

# Buffer limit to X seconds: android





# Buffer limit to X seconds: results

	Android	Web
Zen	- 15% Tb	-
Kinopoisk	-1.5% & -6% traffic/TVT	-

# Dynamic buffer



# Buffer manager

Set of buffer size rules:  $f(t)$ ,  $g(t)$ , ...

$bufferSize = \min(30, f(t), g(t), \dots)$

# Dynamic buffer

Calculate the buffer size from:

- The watched time since start
- The watched time since the last seek position



# Buffer manager

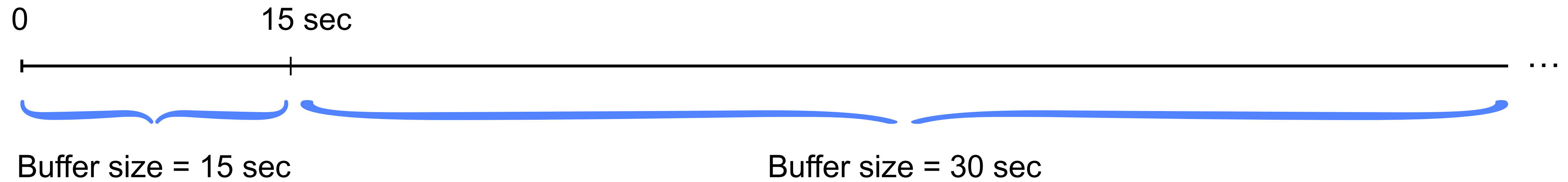
$f(t_w)$  - calculates the buffer size depending on the watched time,  
where  $t_w$  - the watched time in seconds

$g(t_s)$  - calculates the buffer size depending on the last seek position,  
where  $t_s = \text{currentPosition} - \text{lastSeekPosition}$

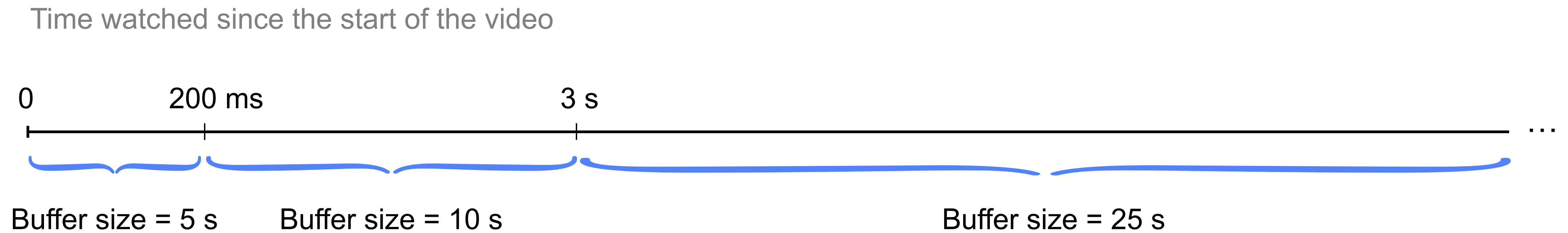
$$\text{bufferSize} = \min(30, f(t_w), g(t_s))$$

# Dynamic buffer relative to the watched time: web

Time watched since the start of the video



# Dynamic buffer relative to the watched time: android

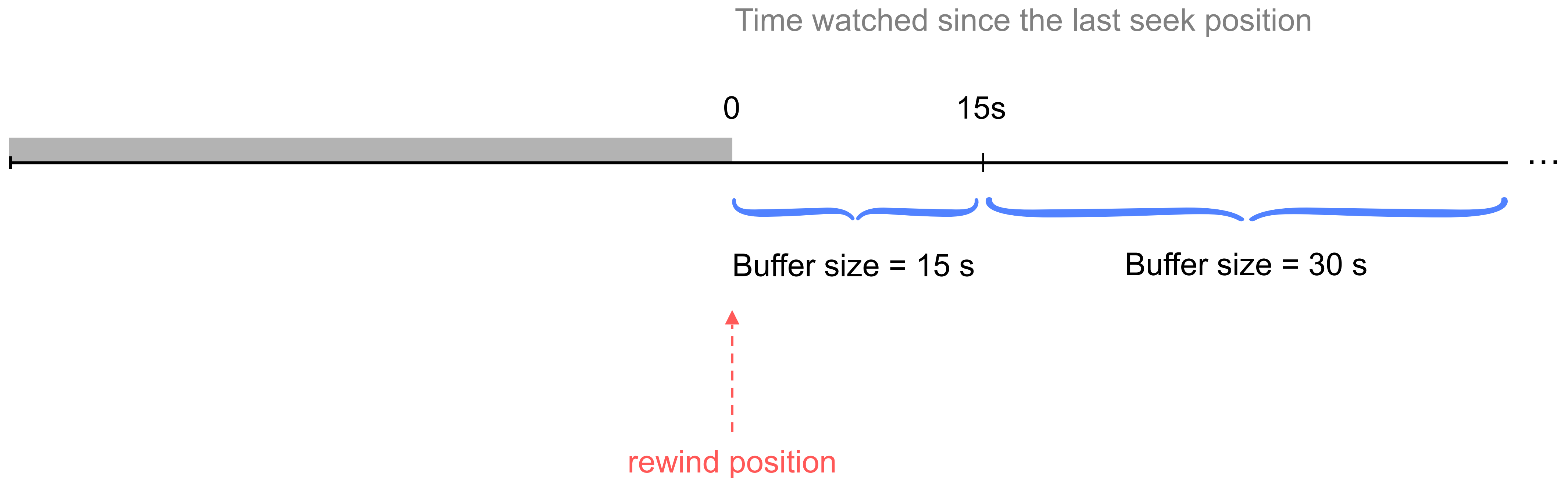




# Dynamic buffer relative to the watched time: results

	Android	Web
Zen	- 21% Tb	- 5 % Tb
Kinopoisk	-	-

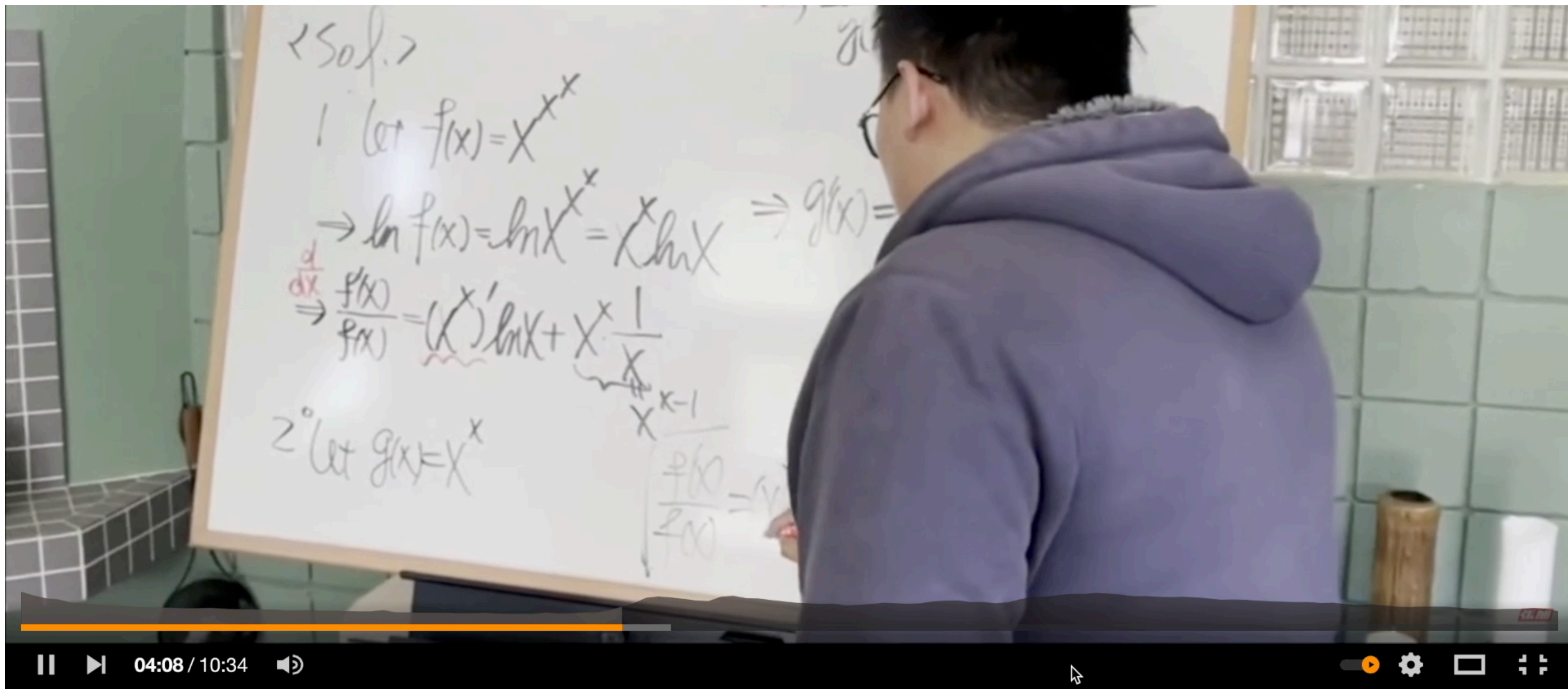
# Dynamic buffer relative to the last seek position: web



# Skippable fragments



# Heat map (most replayed)

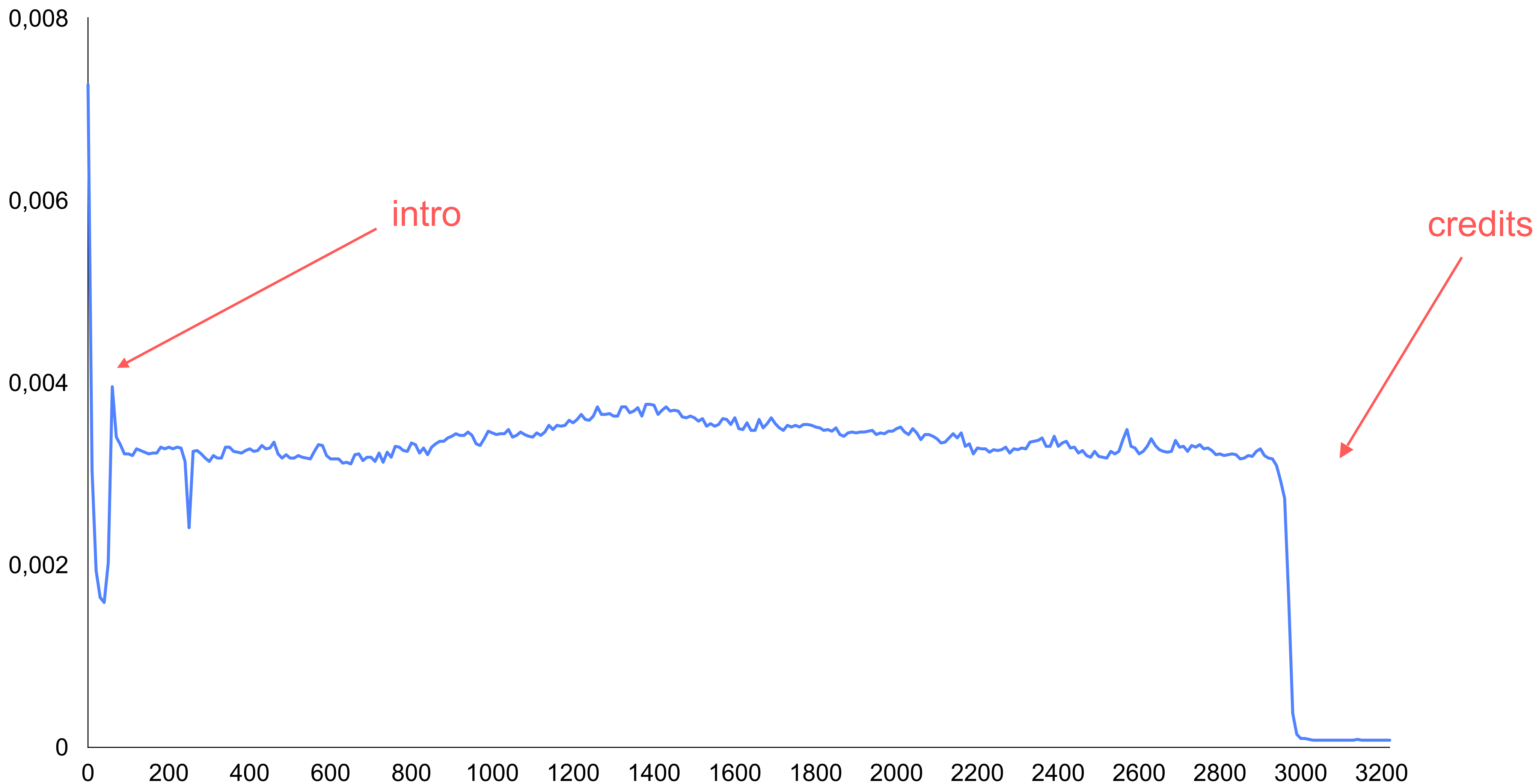


A person wearing a purple hoodie and glasses is seen from the side, writing on a whiteboard. The whiteboard contains handwritten mathematical derivations for the derivative of  $f(x) = x^x$  using logarithmic differentiation. The text on the board includes:

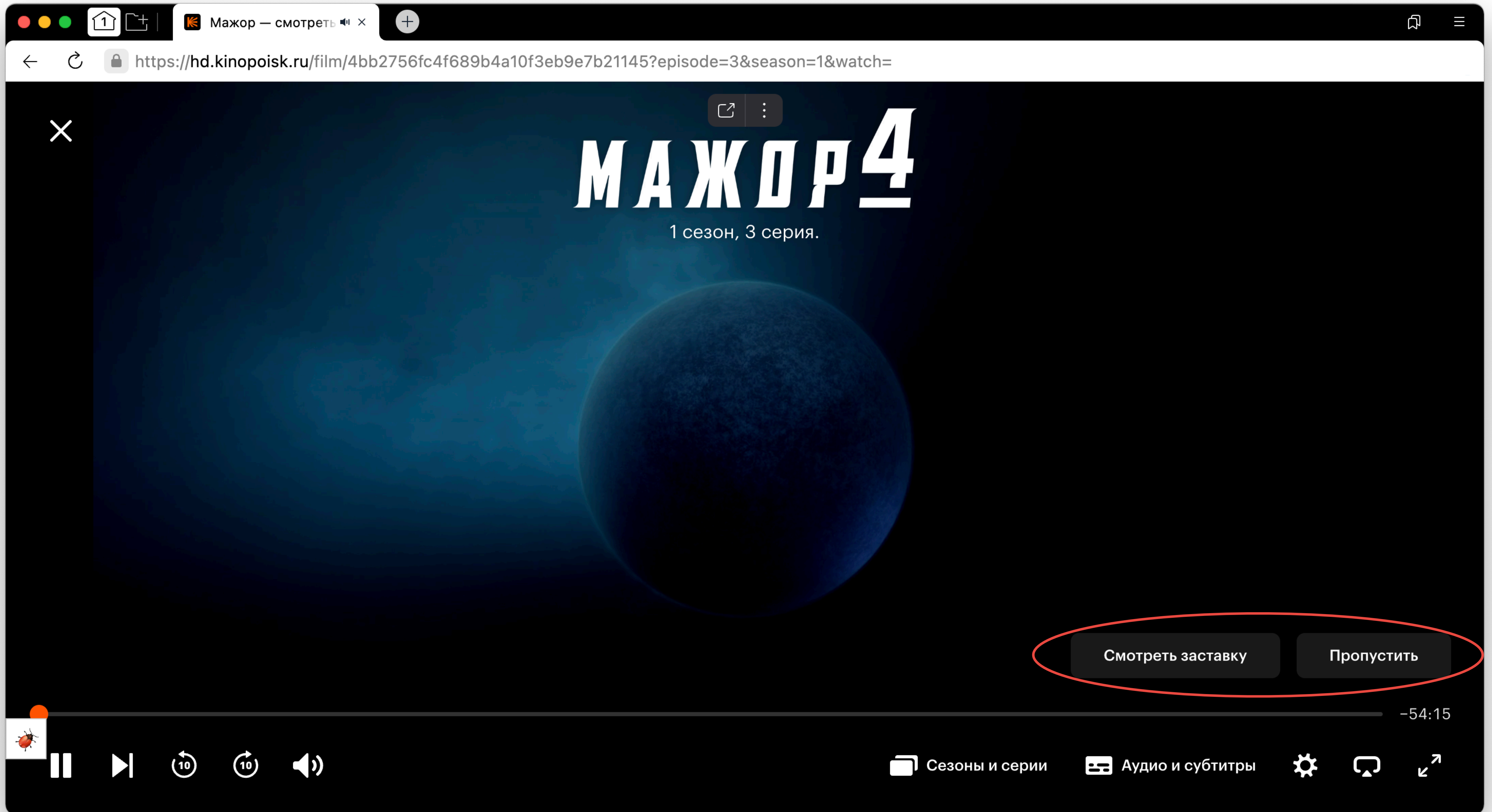
<Sol.>  
1. Let  $f(x) = x^x$   
 $\Rightarrow \ln f(x) = \ln x^x = x \ln x \Rightarrow g(x) =$   
 $\Rightarrow \frac{d}{dx} \frac{f(x)}{f(x)} = \frac{(x^x)'}{x^x} \ln x + x^x \cdot \frac{1}{x}$   
2. Let  $g(x) = x^x$   
 $\frac{f(x)}{f(x)} = \frac{f'(x)}{f(x)}$

The video player interface at the bottom shows a progress bar at 04:08 / 10:34, a volume icon, and a full-screen button.

# Heat map of a some title on Kinopoisk









# Buffer size inside skippable fragments



# Skippable fragments: results

	Android	Web
Zen	-	-
Kinopoisk	-	- 1% traffic/TVT

# Viewport capping

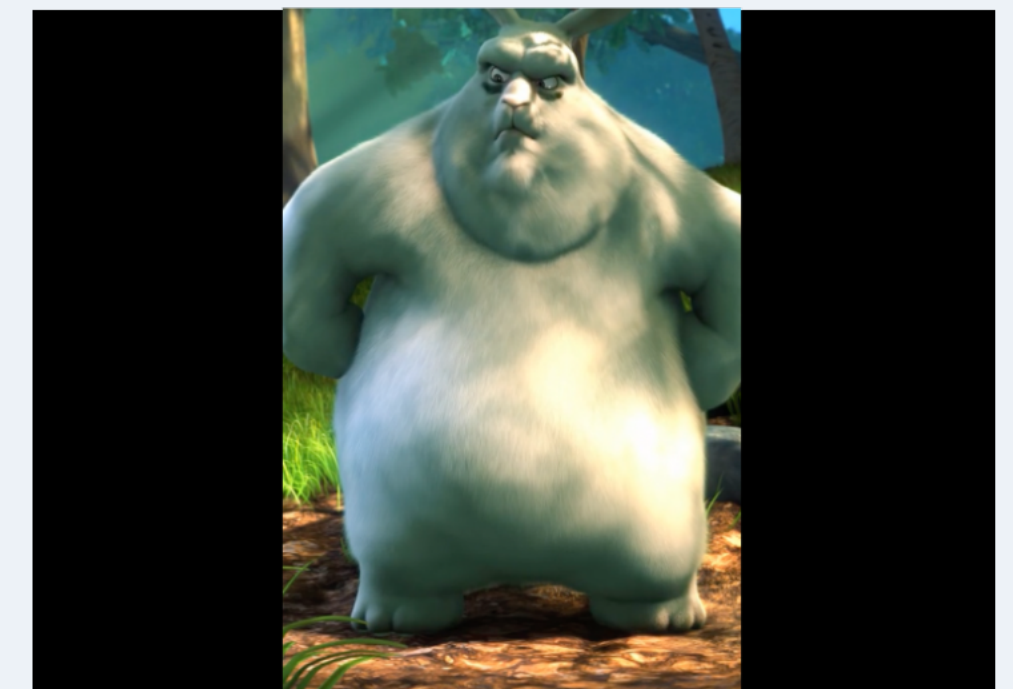
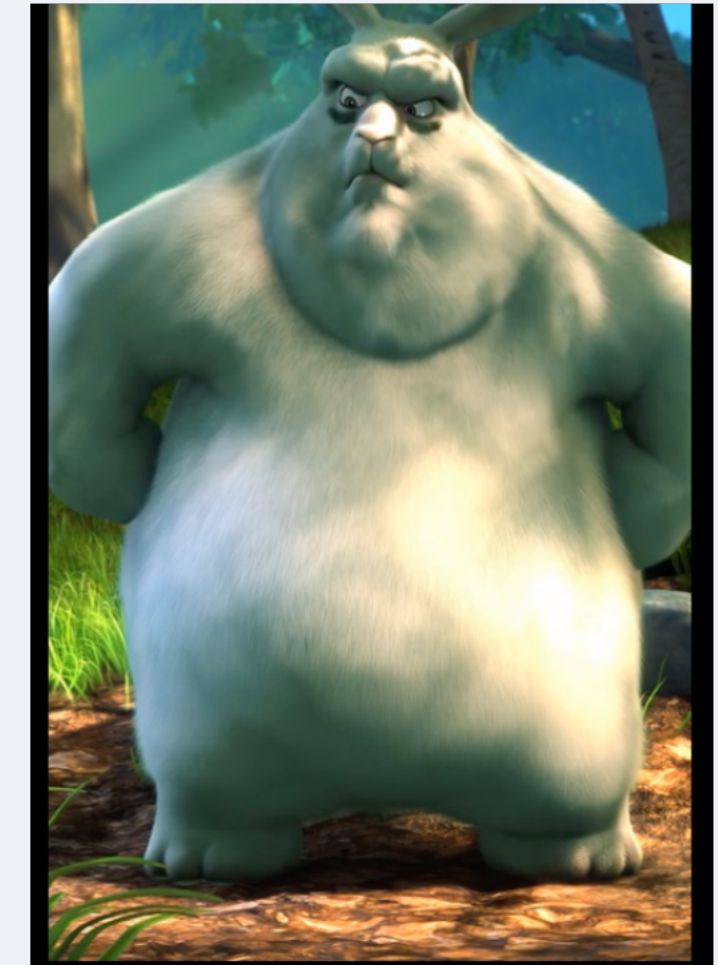


# Capping

Capping - limiting the qualities array according to the container dimensions

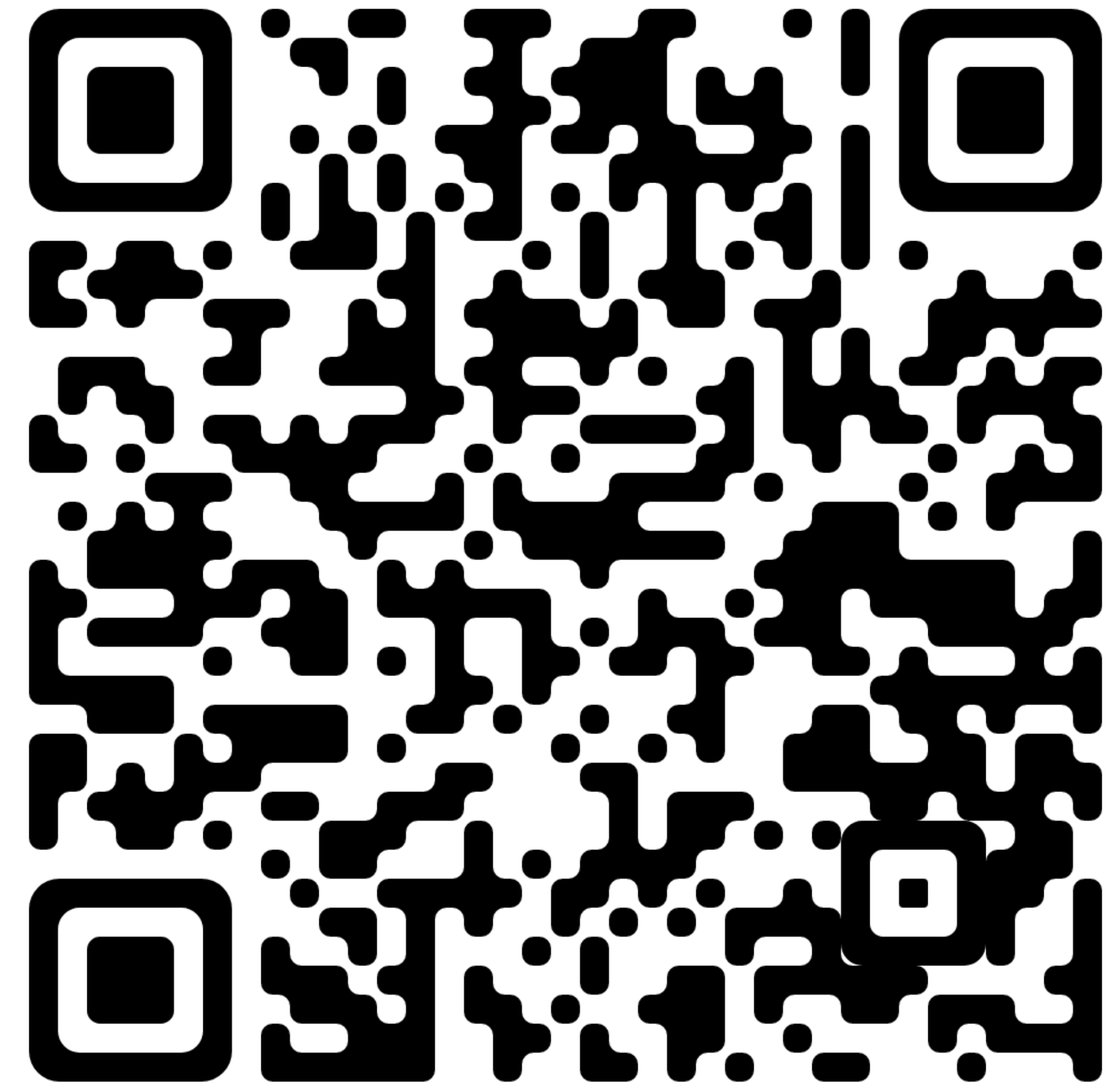
video  
orientation

container orientation



# Capping in hls.js

```
capLevelToPlayerSize: true
```



<https://clck.ru/32qZsi>





## demo v1.2.4

Test your HLS streams in all supported browsers (Chrome/Firefox/IE11/Edge/Safari).

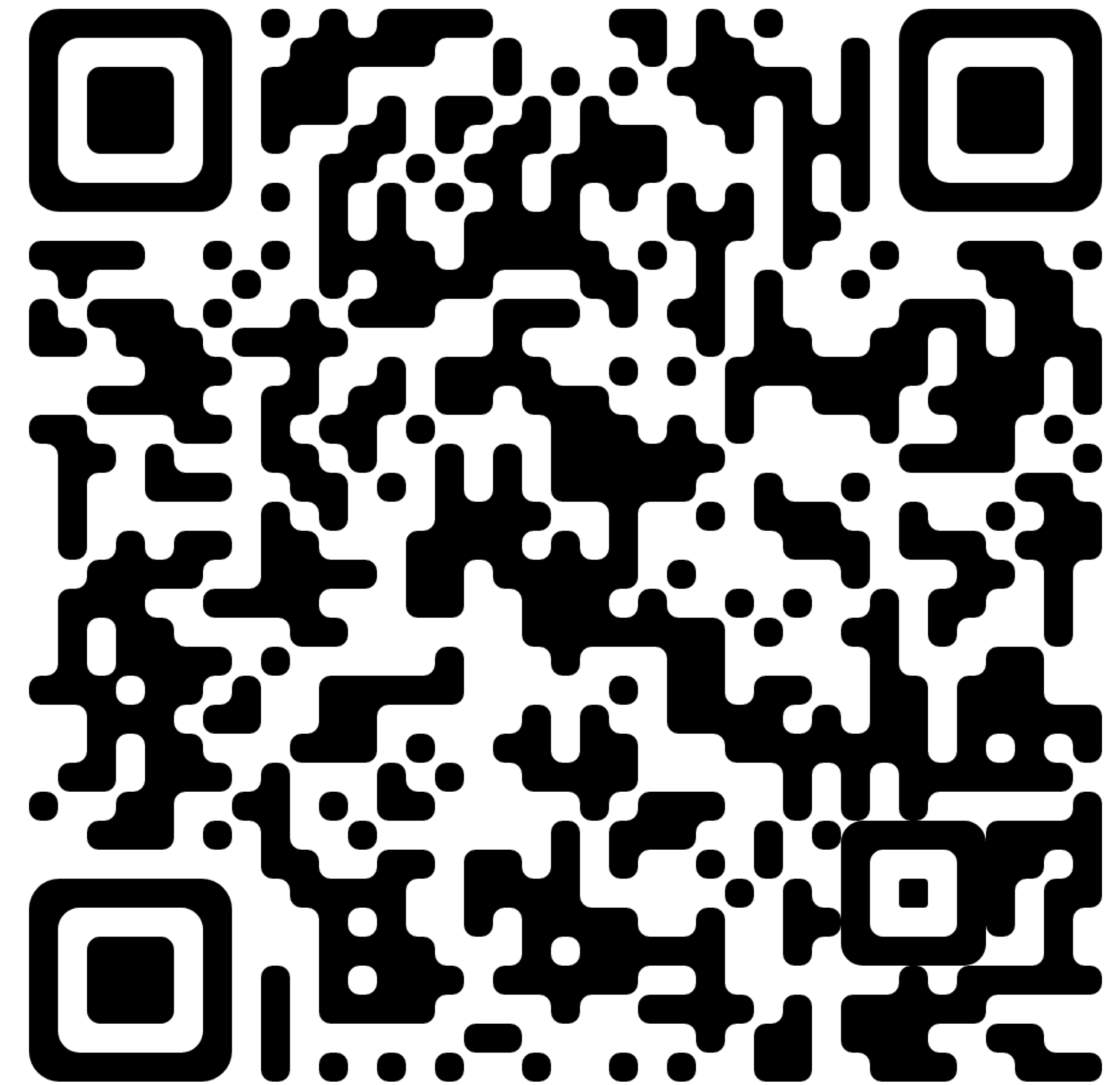
Advanced controls are available at the bottom of this page.

Looking for a more *basic* usage example? Go [here](#).

A wide, horizontal close-up of dense, vibrant green grass and foliage, likely from the movie 'The Lion King'. The image shows a lush, textured field of tall grasses and small plants, with a large, gnarled tree trunk visible on the left side. The lighting is bright and natural, highlighting the various shades of green.

# Capping in dash.js

```
limitBitrateByPortal: true  
usePixelRatioInLimitBitrateByPortal: true
```

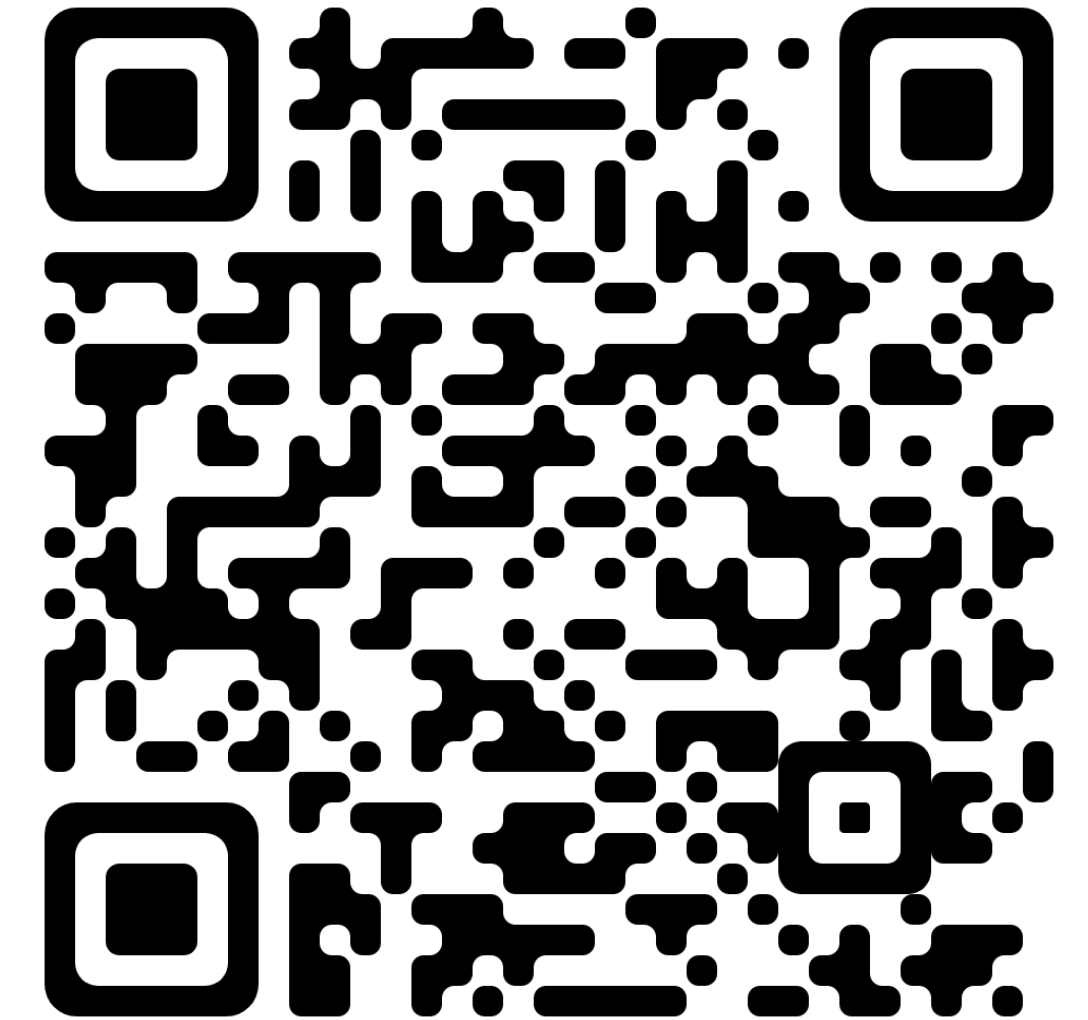


<https://clck.ru/32qZtB>



# Capping in dash.js

```
<script>
  (function() {
    ...
    player.updateSettings({
      streaming: {
        buffer: {
          bufferTimeAtTopQualityLongForm: 20
        }
      }
    });
    ...
  })();
</script>
```



<https://clck.ru/32qZxb>





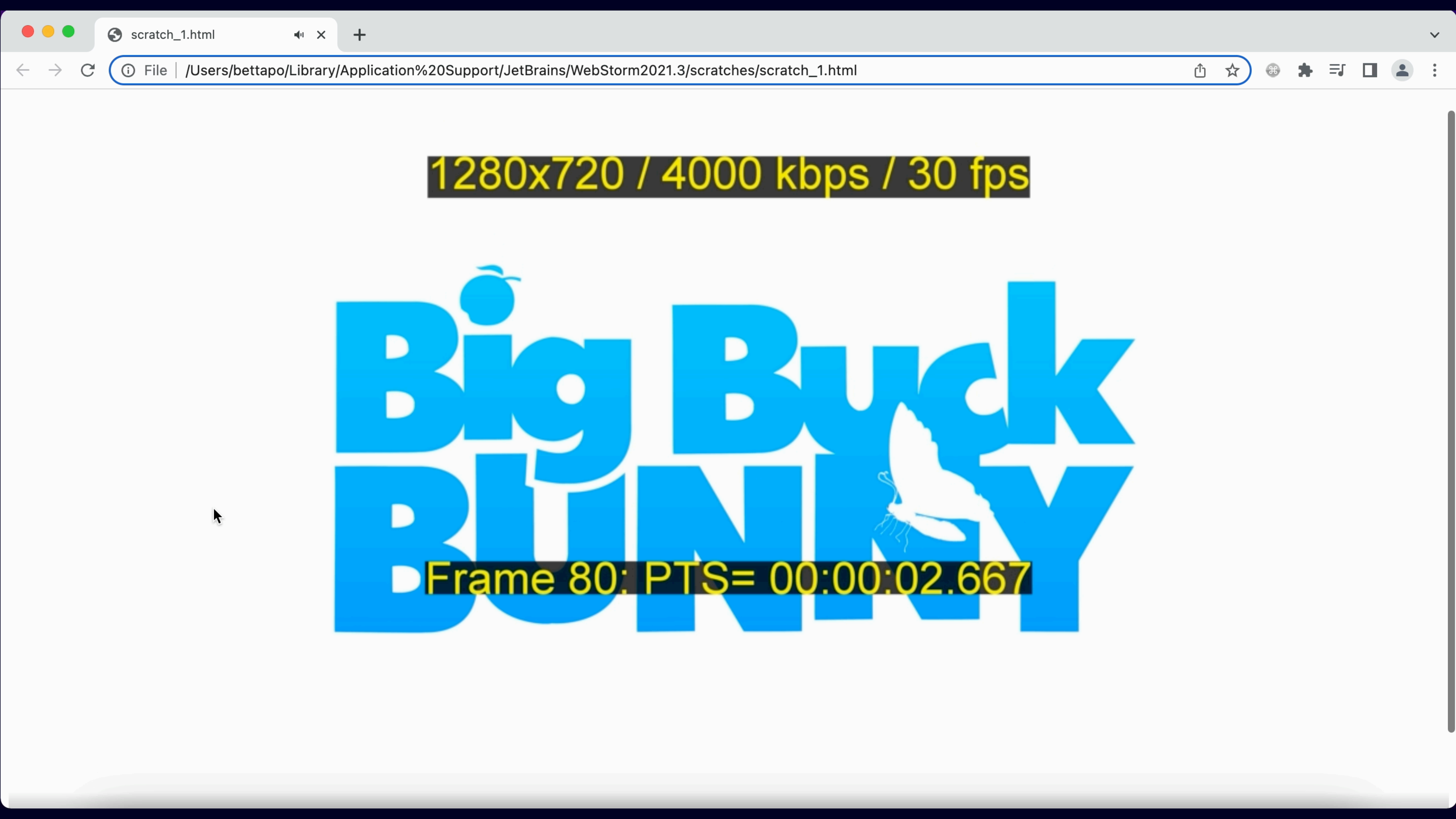
3840x2160 / 12000 kbps / 30 fps

Big Buck  
BUNNY

Frame 39: PTS= 00:00:01.300

# Capping in dash.js

```
<script>
  (function () {
    ...
    player.updateSettings ({
      streaming: {
        abr: {
          limitBitrateByPortal: true
        },
        buffer: {
          bufferTimeAtTopQualityLongForm: 20
        }
      }
    });
    ...
  }) ();
</script>
```



1280x720 / 4000 kbps / 30 fps

Big Buck  
Bunny

Frame 80: PTS= 00:00:02.667



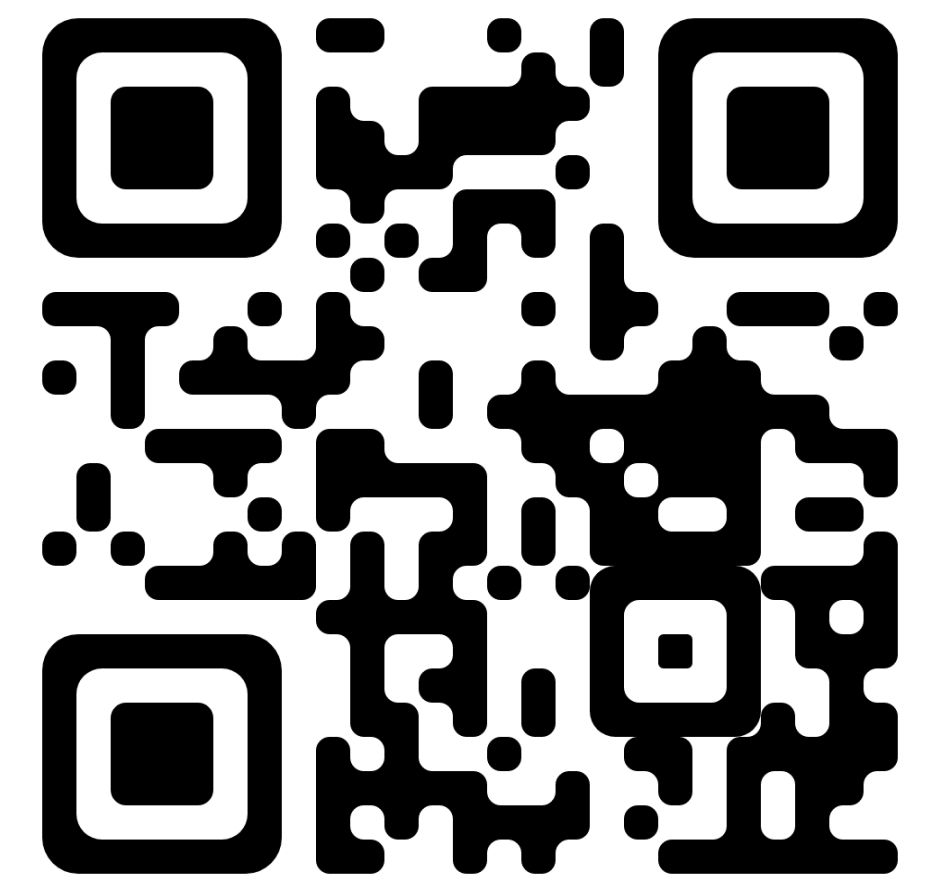
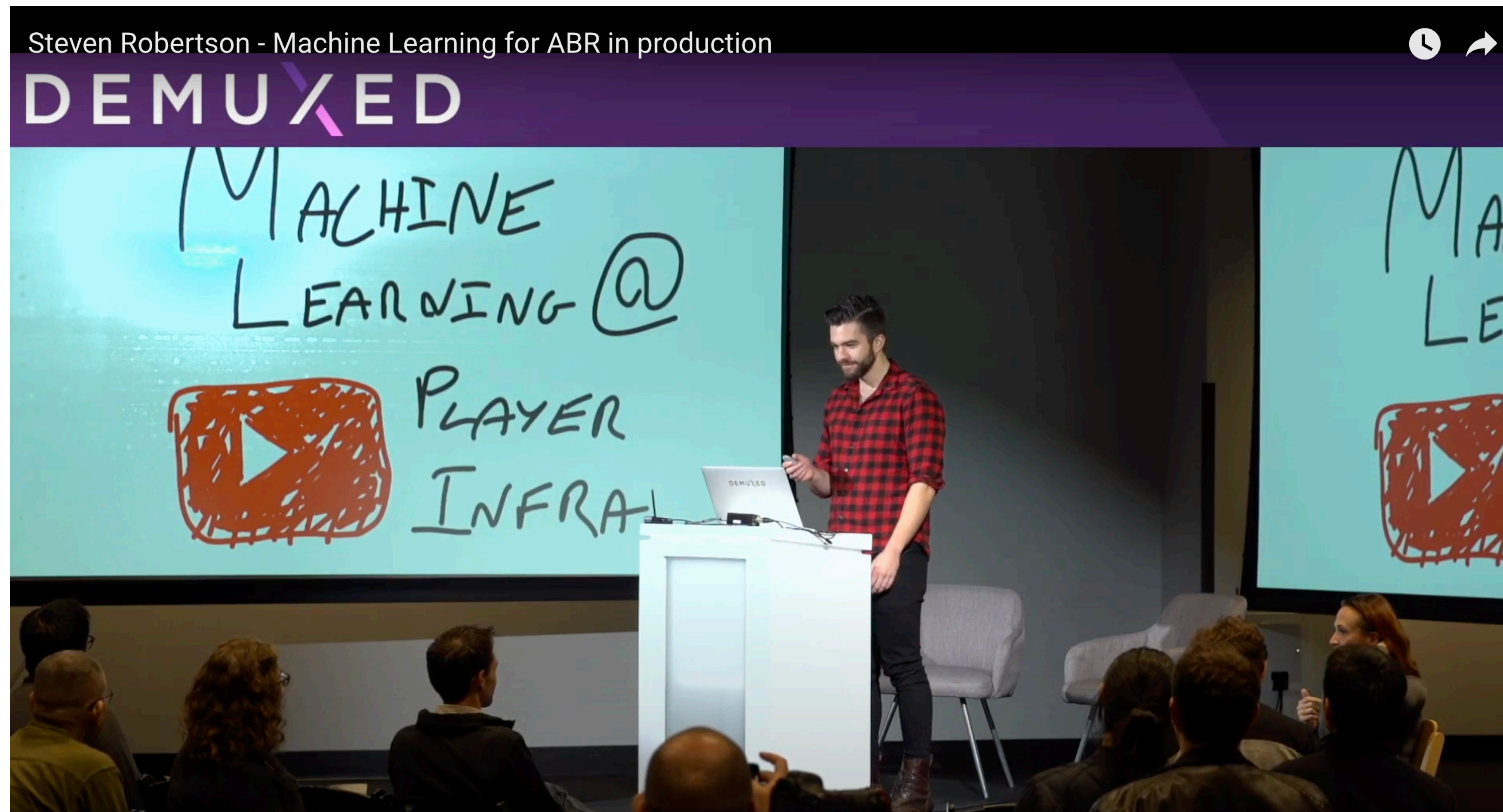
# Viewport capping: results

	Android	Web
Zen	-	-
Kinopoisk	- 5% traffic/TVT	-

# Aesthete capping



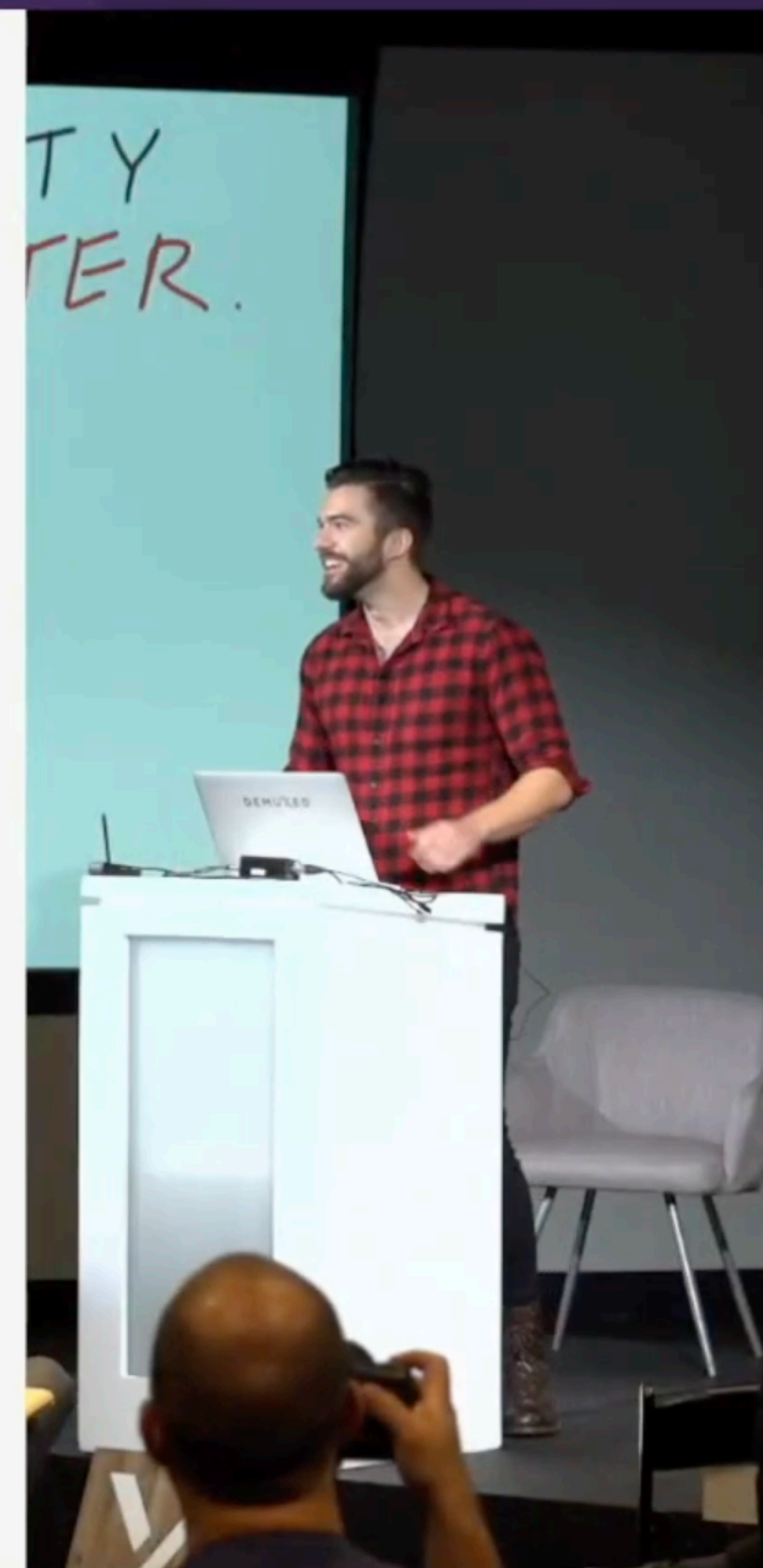
# Steven Robertson - Machine Learning for ABR in production




<https://youtu.be/iuaWhmETcRE>



VIDEO QUALITY  
DOESN'T MATTER.



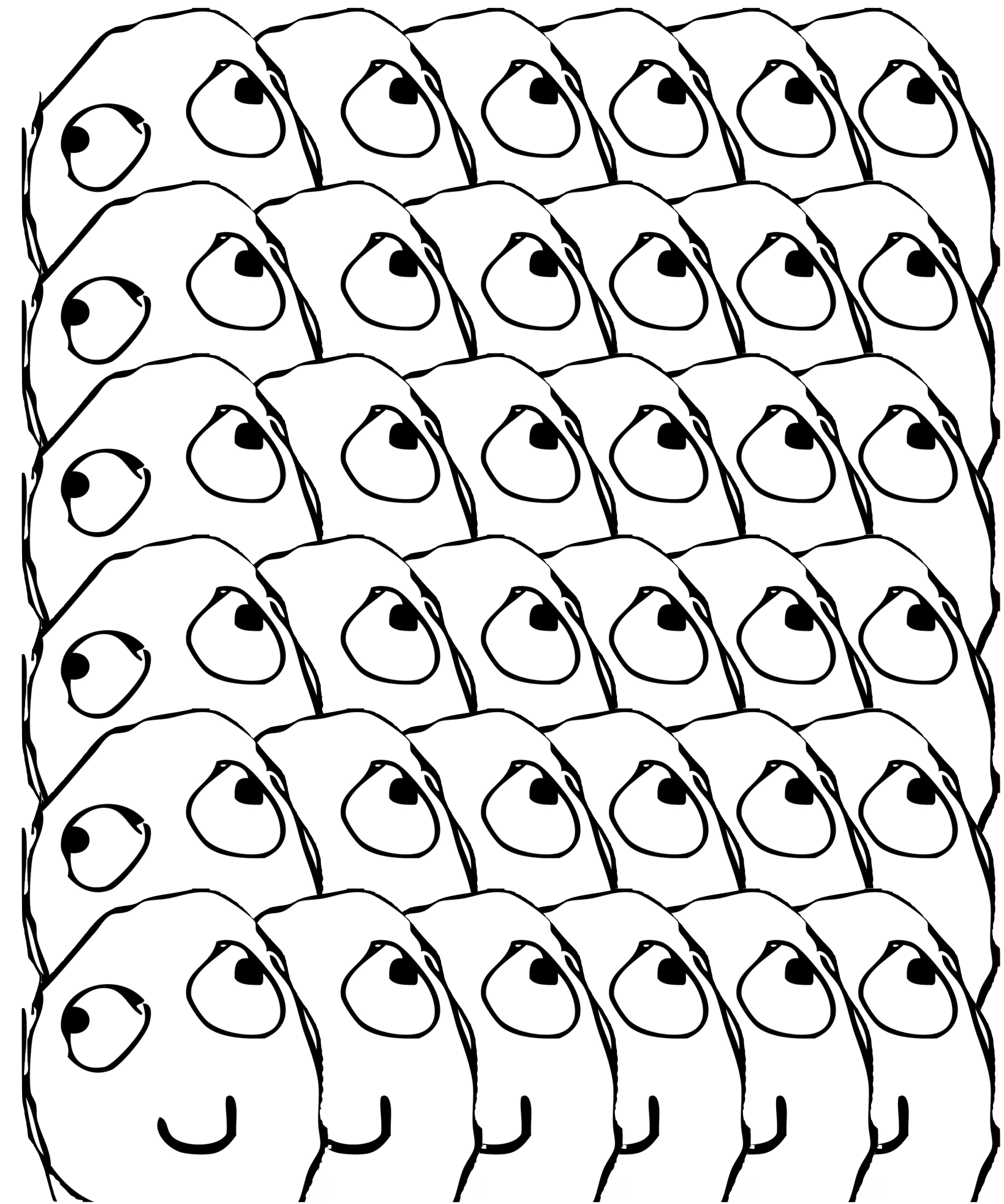


SwitchUp - metric of the frequency of the quality switching up by users

# Users



aesthetes



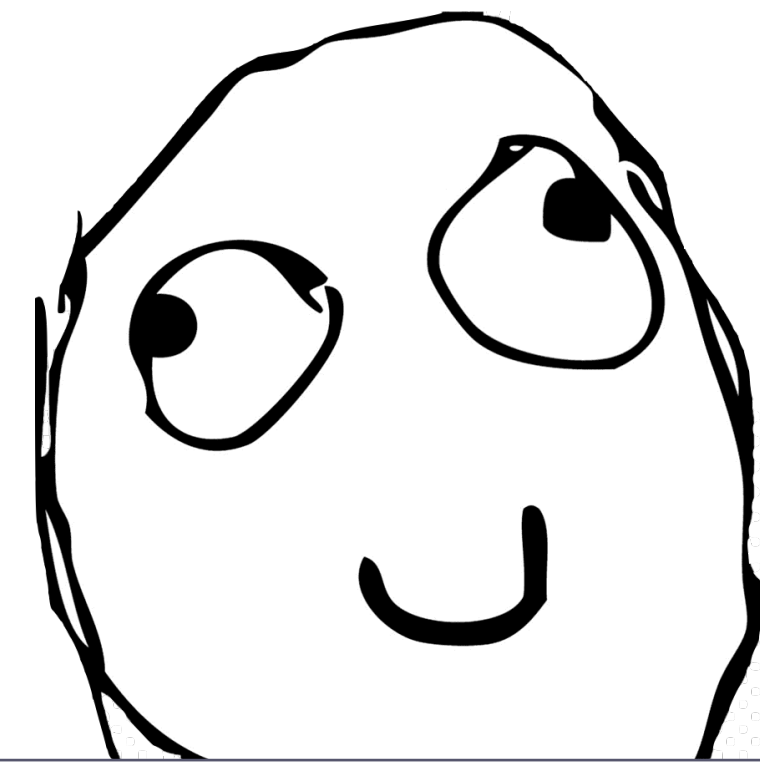
normal users



# Users



no limits



no more than 480p (720p)

# Aesthete: results

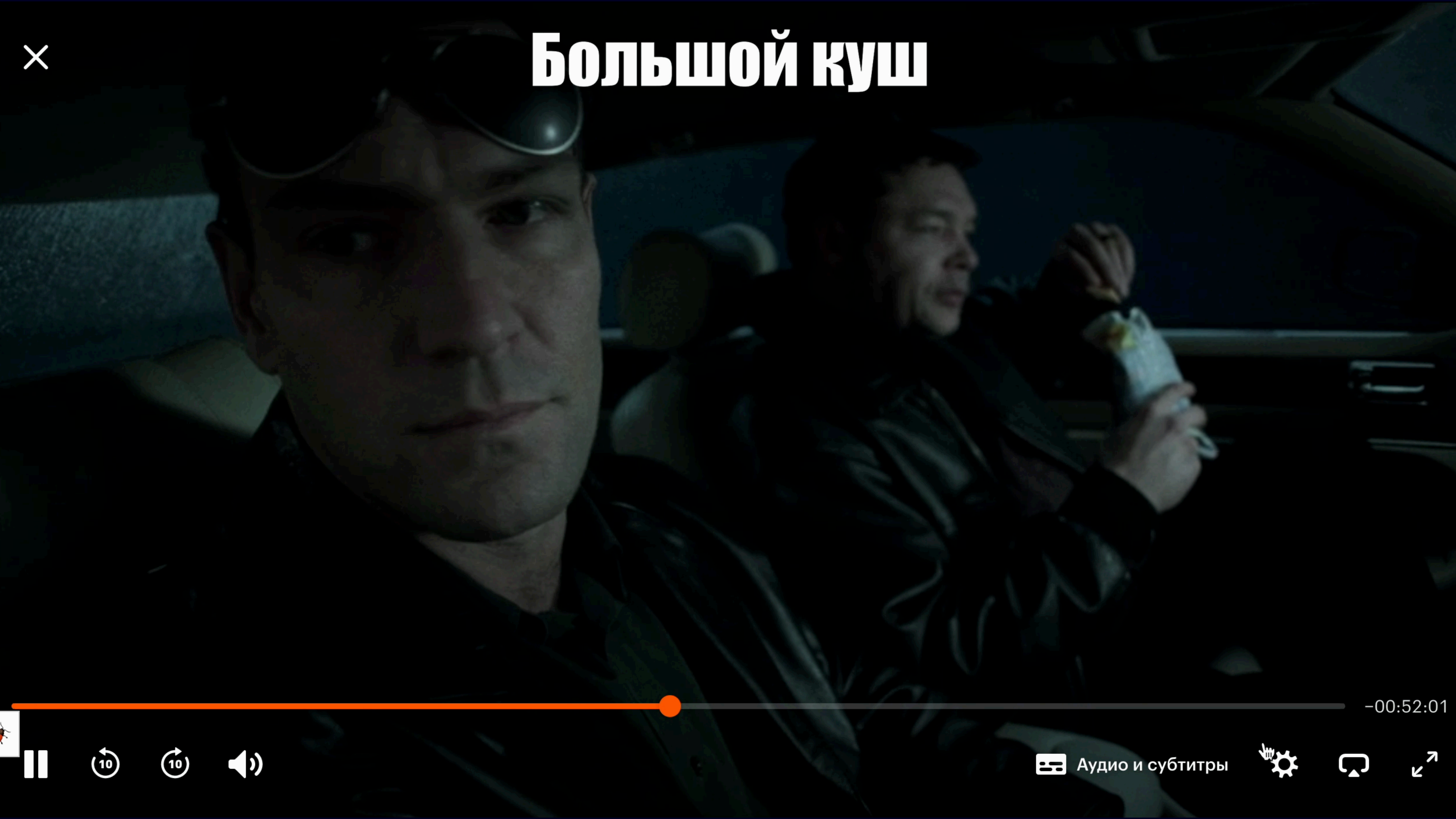
	Android	Web
Zen	-	- 7-10% Tb
Kinopoisk	-40% traffic/TVT	-

# SwitchUp capping





# Большой куш



-00:52:01



Аудио и субтитры



# SwitchUp: results

	Android	Web
Kinopoisk	-	- 4.5% traffic/TVT



# Conclusions





Kinopoisk saves the following % of traffic:

- 4.5% on web
- 39% on android

# Conclusions on hypotheses

- Buffer limit to X seconds
- Dynamic buffer
- Skippable fragments
- Viewport capping
- Aesthete capping
- SwitchUp capping

# Feel free to contact me at any time



**Olga Popova**

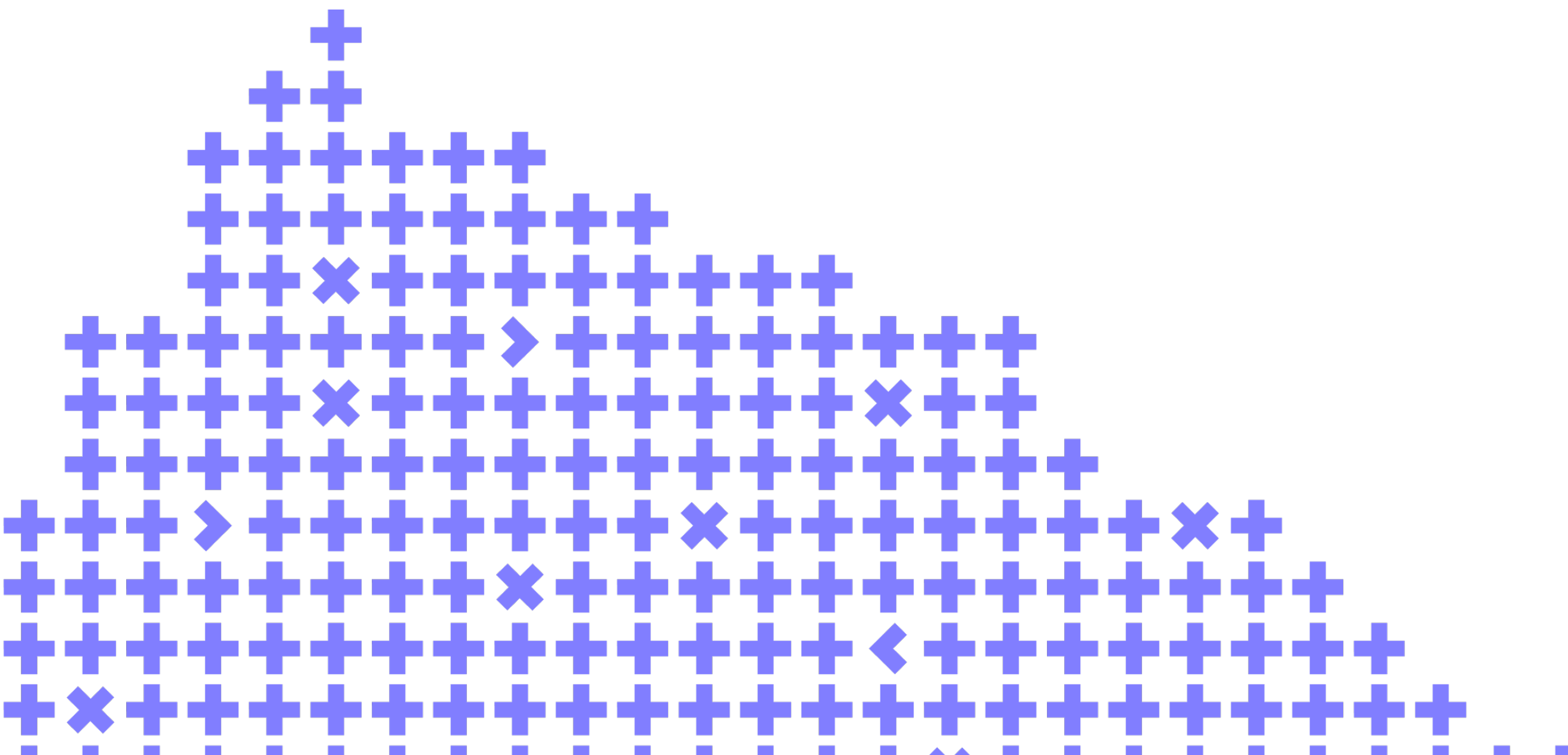
UI developer  
at Yandex Cloud

[bettapo@yandex-team.ru](mailto:bettapo@yandex-team.ru)



Leave your feedback!

You can rate the talk and give feedback on what you've liked or what could be improved



Co-organizer

Yandex